

REMARKS- General

By the above amendment, Applicant has amended and rewritten all claims to define the invention more particularly and distinctly so as to overcome the rejections under 35 USC § 112, 102 and 103, and define the invention patentably over the prior art.

Numbering Errors Corrected

First Error. The previous amendment's claims have a numbering error due to skipping two numbers, that is, claim 177 was followed by claim 180. Therefore, claims 180 through 200 have been renumbered as claims 178 through 198.

Second Error. Claims 167 through 200 are dependent claims. The claim number that each is dependent upon was inadvertently left unaltered after the previous amendment's cancelling of old claims and adding of new ones. The claim number that each is dependent upon has been corrected.

Several claims split, additional claims added at end. Three claims were split into two claims in the interests of clarity (claim 161- split into 161 and 201; claim 182- split into 182 and 202; claim 187- split into 187 and 203). Applicant is aware that dependent claims should be grouped together to the extent possible, however, in the interests of allowing the Examiner to track the changes more easily, the additional dependent claims have simply been added at the end. Applicant expects to renumber the claims to group them together when they are allowed. Applicant believes that additional claim fees are not required because several claims were deleted in this response and the previous response, and the total number of independent and dependent claims is less than the number already paid for.

The Claims Rejections Under 35 USC § 112

The O.A. rejected claims 132-200 because they invoked the 112 sixth paragraph (i.e.: the means).

From the MPEP:

"...the term 'step' alone and the phrase 'steps of' tend to show that §112, 6 does not govern that limitation; *Personalized Media Communications LLC v. ITC*, 161 F.3d 696, 703- 04, 48 USPQ2d 1880, 1886- 87 (Fed. Cir. 1998);"

Claim 132-135, 138-140, 143, 148, 150 contain the phrase "steps of" and do not contain the words "means for" or "step for". Functional language is not used. They contain sufficient functional language to achieve the specified function. Therefore, according to the MPEP, § 112, 6 does not govern those limitations.

The other claims have been rewritten to replace means-plus-function language with the corresponding structure.

Claim 136 and 137 were modified by adding the phrase "steps of". Claim 141 is a device claim, and does not contain the words "means for" or "step for". The words "capable of" in claim 141 para (f) were replaced by the word "that". Claim 142 does not contain the words "means for" or "step for". The word "means" was removed from claim 143 paras d-h and claim 144 paras a-b and replaced with "computer software executable code". Claim 145 deleted the words "a first communications means consisting of". Claim 146 para a replaced the words "communications means" with "communications program or application", and in para b replaced "address configuration means" with "address configuration computer executable code". Claim 147 replaced the word "means" with "program or application". Claim 149 replaced the word "means" with "program". Claim 151 paras d-i, replaced the word "means" with "computer

software executable code". Claim 152, replaced "hyperlink generation means" with "computer software executable code". Claim 153, deleted the words "providing hyperlink transmission means for". Claims 154-160, being dependent on claim 151 or 152, are now not applicable to § 112, 6. Claim 160, "means" replaced with "computer software executable code". Claim 161 was divided into two claims, with the first claiming the basic file access and manipulation restrictions, and where "file access and manipulation restriction means" was replaced with "computer software executable code". The second claim, Claim 161a, dependent on Claim 161, adds variability to the file access restrictions. Claim 162-163, "file restriction means" replaced with "computer software executable code".

The Claims Rejections Under 35 USC § 102 (e)

Claims 132-135, 141, 142, 151-163, 171, 177, 180-182, and 187-195 were rejected under USC § 102 as anticipated by Simonoff et al (Simonoff, 6,005,568). Applicant requests reconsideration and withdrawal of this Reason for the listed claims because the element shown in the prior art (Simonoff, 6,005,568) is not an equivalent of the structure, material or acts disclosed in the application. The listed claims recite structure that is physically different from every cited reference and is unobvious under USC § 103.

Simonoff Prior Art Summary. The cited patent is essentially for a script interpreter running on a client computer, which sits on top of a Virtual Machine (VM) such as Java. This script interpreter, called a Universal Client (UC), generates a graphical user interface (GUI) on the client computer as directed by a script embedded in (or referenced by) a web page [Simonoff, utilize a virtual machine... to generate predetermined GUI objects and display the GUI objects on the computer, col. 6, lines 8-13]. This GUI consists of buttons, scrollbars, and other graphical elements contained in a library on the client computer.

The Rejection of Claim 132

Physical Differences Over Simonoff

Thin Client Physically Different Than Simonoff. The meaning of the term "thin client", as known in the art, refers to a simple client program or device which relies on most of the functionality of the system being located on the server.

"A term used in the claims may be given a special meaning in the description, but no term may be given a meaning repugnant to the usual meaning of the term."
MPEP 608.01(o); 2173.05(a).

The applicant's more specific, distinct, and non-repugnant meaning of "thin client", as described in the specifications, is wherein the GUI is *generated or created* on the server and transmitted and displayed to the user via the thin client, in essentially a *terminal emulation* mode. The major processing done by the thin client is to send user input (keystrokes and mouse movements) to the server and receive the updated GUI state from the server and display it on the user's screen. Examples of thin clients of this type known in the art include Citrix and Tarantella. These major commercial products demonstrate clearly that the applicant's definition is within the meaning of the art.

Definition of Thin Client From Specifications. An excerpt from the current application's specifications defining "thin client" make clear the above physically different meaning of the term:

"A definition of thin client as intended by this invention may be useful. This terminal emulation may proceed by input at the client machine being transmitted as instruction requests to the server. Upon execution of the instruction request at the server application level, the server web-enabling application may modify the

GUI or other interface state of the remote user's data file, as returned by the "user application" running on the server, i.e., the application that the remote user wishes to use for manipulation of a certain data file. In other words, the user's input on the client machine is transmitted to the server, executed on the user application running on the server, and *the output of the user application is sent to the client machine, basically, as a "picture" of the server's executable user application output.*

Because this picture is preferably changed very frequently, the experience of the client user is substantially the same as if the user was inputting commands to executable code resident on the client machine. This "picture" repeatedly sent to the client user may be termed the "GUI state" of the application executing on the server. The executable code which allows the client machine to emulate a terminal, i.e., allows the client machine to send instructions to the server machine, and display the GUI or other output of the client machine, may be considered a component of enable code, and is called commonly a "thin-client". If a client machine already has the *terminal emulation thin client software* installed, the enable code downloaded to the client machine may consist solely of the server-based application's "GUI State". [Substitute specifications, page 9, lines 22-31, page 10, lines 1-8].

Physical Difference- Claimed Invention's GUI is Created on Server, while Simonoff's is Created on the Client. In Simonoff's Universal Client applet, *the GUI is generated on the user's local computer.* In contrast, in the claimed invention, the GUI is created on the server, and the response to user input is defined by the server-based application, not a script running locally.

Simonoff teaches that instructions on how to generate a GUI are created on the server or

embedded into a web page, but the actual display generation happens on the client, according to GUIscript instructions downloaded from the server. In contrast, in the claimed invention, the GUI is generated *on the server by the remote application*. In the present invention, no instructions on how to generate a GUI are created on the server.

In fact, Simonoff's main teaching is how to create user GUIs on the client [Simonoff, an object of the present invention is to create user front end GUIs...col 5 line 5], whereas the present invention does not teach how to do this, because there is no need: The GUI is already generated by the native application running on the server.

Simonoff teaches how to create a GUI on the client, while the present invention claims a method including the transmission of a GUI to the client in essentially an instantaneous terminal emulation mode.

Physical Difference- User Input to GUI Handled Differently. The response to user input in Simonoff is defined by the downloaded script:

"Any number of actions can be performed responsive to the operation of a GUI button, i.e., when the button is clicked. The actions, called events, are defined in the GUIScript language", Simonoff, col 11 lines 55-58.

As can easily be seen by this citation, under Simonoff user input is handled on the client PC by the Universal Client. In contrast, according to the current invention, user input is transmitted unaltered to the server, and the response happens on the server by the application program.

Physical Difference- Simonoff Cannot Function According to Present Invention. If

no instructions on how to create a GUI are downloaded to the client, as claimed in the present invention, Simonoff's invention cannot function, since there would be no GUI for a user to interact with. Nowhere in the cited reference does Simonoff teach the use of a simple terminal remote interface, as claimed by the present invention.

Physical Difference- With Simonoff, Applications With No GUI Can Be Accessed.

Simonoff also teaches that server-based applications (such as databases) that have no native GUI can be accessed by using a GUI which is generated on the client. [Simonoff, the UC device responds in a predetermined way when a character string ...is received. Thus, the UC device can process information from a data base application, col 11 lines 26-30] In contrast, with the present invention, a database file would be accessed by operating on the server a database client application compatible with the file *and* capable of generating a GUI. Interfacing directly with a remote application that does not generate its own GUI is not possible with the current invention.

Physical Difference- Simonoff Does Not Associate A File And Online Application With A Hyperlink. This feature is the essence of the claimed invention, and nowhere in Simonoff does he explicitly link the three.

Physical Differences Unobvious Under USC § 103

Valuable, New, Improved, and Unexpected Results. The differences detailed above are *unobvious differences* under § 103 because they result in valuable new, improved, and unexpected results. A few of those, as disclosed in the specifications, are:

Improved Access To Applications And Files. Associating a hyperlink with an online

application and file and giving access to said application and file via a thin-client delivered GUI allows the easy, simple access to remote applications and files to be greatly expanded over the art at the time of invention. Specifically, applications and files can easily be added to emails, web pages, or any other hyperlink-capable communications medium. For example, an ordinary web page can giving a user access to a dozen different applications, running on three different operating systems. This access can be done simply by pasting into the web page a few URLs. Or instead of attaching a file to an email, an AppLink™ to the file can be pasted into the email, thus bypassing any restrictions on file attachment size by the recipient's email service. Additionally, there is no need for the recipient to have a compatible application installed on his local PC. For example, a hyperlink to a Microsoft PowerPoint file can be sent, and the recipient can view and interact with the presentation with the full power of the native application. This is far superior in capabilities to the alternative of using some sort of downloadable file viewer or browser plug-in.

Improved Data Security. Simonoff's invention does not keep data secure by keeping it on the server. In fact, Simonoff teaches that raw data is transmitted to the client after packaging in a GUIScript message [Simonoff, Figure 1, Datagram feeds into GUIscript message]. Security is provided only thorough secondary means, like providing a secure network. [Simonoff, the computer sustem 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. This is in contrast to the claimed invention, where security is intrinsic to the invention because raw application data is not transmitted to the client. Only GUI refresh data is transmitted. A thin client as claimed keeps the applications *and data* remote from the user. Only the GUI screen or output is transmitted to the client computer.

Improved Intellectual Property Protection. Because the file and application remain remote from the user, many new and unexpected capabilities to protect the file become possible versus the alternative of downloading the file to the user. Some are: Restricting

the time that a file can be accessed (such as restricting a test to be taken to a one-hour period, ____.

No GUI Programming Required. Simonoff's GUI is generated by the client computer, and is specified by the script (called a GUIscript). The GUI is generated by the user's local computer. The programmer must design each element of the GUI and specify the actions done. This must be done for each script [Simonoff, any number of actions may be performed responsive to the operation of a GUI-button...The actions, called events, are defined in the GUIScript language, col 11 lines 54-58]. This is in contrast to the claimed invention, where the GUI is generated by the application remote from the user, with no programming required.

Not Limited by Client Graphics Toolkit. In the claimed invention, the GUI is not limited by the graphics toolkit on the client computer because it is generated remotely. Whereas Simonoff *is* so limited [Simonoff, GUIscript can instantiate any GUI object common between Microsoft Windows, X-Windows, and the Java AWT, col. 11, lines 41-43].

Claim 132 Meets Novelty and Unobviousness Requirements of § 102 and § 103. As abundantly detailed above, claim 132 has physical differences over Simonoff that are unobvious to one with ordinary skill in the art. Specific Reasons raised by the O.A. are discussed next.

The Rejection of Claim 132 Is Overcome

Claim 132 was rejected because the Office Action stated that Simonoff discloses a method to allow a user of a local computer to access a computer file, the method comprising the steps of:

- (a) providing a communications network [Simonoff, WAN, Internet, Fig 3];
- (b) detecting over a communications network the activation by the user of a hyperlink associated with the computer file [Simonoff, read, parse and process the reference in HTML

code, col 10 lines 12-27; URL, col 9, line 67; filtering or filtered, col 10, lines 41-55, 56-col 11 line 6].

Simonoff "the reference in HTML code" Is Not A Hyperlink. The Simonoff citation does not refer to detecting activation of a hyperlink or URL. The word "reference" refers to GUIScript commands, and to the client applet reading and executing the GUIScript commands embedded in a HTML document to create a GUI.

Incomplete Citation. The complete quote from Simonoff shows that the word "referenced" is simply an alternate form of the word "embedded", and is not a URL:

[Simonoff, the Universal Client device advantageously can read, parse and process the GUIScript commands embedded or referenced in the HTML code of the web page containing the Applet tag for the Universal Client device, col 10 lines 17-20].

Simonoff URL Doesn't Point To A File- Physical Distinction. The cited Simonoff reference "URL" refers to a hyperlink pointing to the server *host*, and not to a particular computer file [Simonoff, the GUIScript downloaded from server host 100a includes the URL pointing to server host 100n of fig. 2, col 9 lines 66-67].

Simonoff "filtering" Is Not Hyperlink Detection: . The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

(c) operating on a computer remote from said user an application program compatible with or capable of loading and operating upon the computer file [Simonoff, a format compatible, col 9 lines 10-29, remote location, col 16 line 63]

Simonoff "format compatible" Refers to Application Running on Client, Not Server. The Simonoff citation "a format compatible" does not refer to a remotely-operating *application*

compatible with a computer file, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23].

Nowhere in Simonoff is the word "compatible" associated with a remote data file, because the Simonoff invention is application-oriented, not data file-oriented.

(d) in response to the detection of the hyperlink, opening the computer file in the application program running on the remote computer [Simonoff, loads, initializes and runs, col 8 lines 26-50; remote location, col 16, lines 63].

Simonoff "loads initializes and runs" Happens On Client, Not Server. The Simonoff citation "loads, initializes and runs" does not refer to an application program operating on a server. Rather, it refers to the Universal Client applet *running on a user's PC* [Simonoff, "...the Universal Client device is to be downloaded to the user's computer...after the Universal Client device loads, it initializes and runs" col 8 lines 47-50].

Simonoff "remote location" Is For Data Storage, Not Operating An Application. The Simonoff citation "remote location" refers to storing user information, such as screen size, remotely, i.e., on the server, and not to operating an application program compatible with or capable of loading the computer file. [Simonoff, , information on each user such as screen preferences...may be stored at a remote location, e.g., server host 100, col 16 lines 62-64].

(f) operating a thin client on the local computer, the thin client allowing the user to provide input to and receive output from the application program running on the remote computer.. [Simonoff, thin client, col 2 lines 45-62; intranet and security reasons, col 8 lines 1-15; server based applications, col 15 line 55-col 16 line 11].

Simonoff "thin client" Generates GUI On Client, Not Server. As detailed above under general remarks, Simonoff's thin client, which generates and operates the user's GUI on the client

PC, is not an equivalent of the claimed invention, where the user's GUI is generated by a remote application and transmitted to the thin client in essentially a terminal emulation mode.

Claim 132 has been rewritten to incorporate the definition of thin client as detailed in the specifications.

Simonoff Sends File Data To Client, Opposite To Claimed Invention. The Simonoff citation "intranet and security reasons" is a reference to achieving data security by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. Simonoff does not retain application file data on the server as claimed. In fact, Simonoff teaches that actual file data (not screen refresh data, as in the claimed invention) is downloaded to the client [Simonoff, "an application... can generate data which will eventually arrive at the Universal Client device on the client host in the form of an incoming GUIScript message", col 12, lines 42-43]. In contrast, with the claimed invention security is intrinsic and does not depend on the network, because the native data is retained on the server and is never downloaded, even in part, to the client.

Simonoff "server based applications" Do Not Operate GUIs. Simonoff teaches that the GUI is created on the client PC by the Universal Client applet. Thus he does not mention any GUI which is intrinsic to, or already a part of, the networked (server-based) application [Simonoff, method for creating user front end GUIs for networked applications, col 5 lines 16-36]. This functions oppositely to the claimed invention where the server-based application must have a GUI which is transmitted to the user.

Claim 132 has been rewritten to include the requirement that the remote application have a GUI.

The Rejection of Claim 133 Is Overcome

Claim 133 Is Dependent on Claim 132, and so meets Requirements of § 102 and § 103.

Claim 133 is dependent upon, and incorporates all of the limitations of, claim 132. Therefore, since claim 132 meets the novelty and unobviousness requirements, so also does claim 133. Specific reasons for the rejection of claim 133 raised by the O.A. are discussed next.

First Reason. The first reason for rejection of Claim 133 stated that Simonoff discloses the hyperlink contains a unique identifier (i.e.: html tags).

A Hyperlink Cannot Contain An HTML Tag. HTML documents are text files made up of HTML elements. HTML elements (such as text) are defined or formatted using HTML tags. The HREF HTML tag can contain a hyperlink, i.e. , but a URL cannot contain an html tag.

Claim Rewritten. Claim 133 has been rewritten to delete the reference to a unique identifier: The claim as rewritten now reads:

Claim 133. (NEW) (old claim 120) The method of claim 132, wherein the hyperlink ~~contains a unique identifier, and further wherein the method includes the step of associating the unique identifier with meta data identifying the computer file.~~ points to a network location or address associated with metadata identifying the computer file.

This rewriting does not add new matter because references are contained in the specifications which interchangeably refer to a unique hyperlink designation, a URL location, and a network location or address:

"...which generates at 610 a unique hyperlink designation, e.g., a URL location over an IP network such as the Internet," Specifications, page 22, 2nd paragraph;
"A method is provided by which a user, e.g. a remote user, may access remote

files and applications in thin client mode by accessing a hyperlink to a URL or network location." Specifications, Summary of the Invention, 1st sentence; "He may then create an actual AppLink or network address or URL at 3504." Specifications, page 31, 2nd paragraph.

Second Reason. The second reason for Office Action rejection of Claim 133 stated that Simonoff discloses associating the unique identifier (now network address) with metadata identifying the computer file as an inherent feature of HTML files.

Associating An Identifier (Now Network Address) With Metadata Identifying the Computer File Is Not An Inherent Feature Of An HTML File. Even if the unique identifier is considered to be a URL, it does not contain and is not associated with file-related metadata.

HTML Document Is Not Metadata. The HTML document pointed to by the URL may have HTML tags, but the tags are not metadata. An HTML tag is not metadata because it is *part of* the file (HTML document), not *about* the file.

Definition of Metadata As Known In The Art: Metadata literally means "data about data". or information that describes another set of data. A common example is a library catalog card, which contains data about the contents and location of a book: It is data about the data in the book referred to by the card. Other common contents of metadata include the source or author of the described dataset, how it be accessed, and its limitations. Nearly all file systems keep metadata about files out-of-band (that is, separate from the file itself). Some systems keep metadata in directory entries; others in specialized structure like inodes or even in the name of a file. Metadata can range from simple timestamps, mode bits, and other special-purpose information, to icons and free-text comments.

Physical Difference With Known Art. Combining the step of associating a URL with metadata which links the URL with a particular file with subsequent steps of opening the file in a server-based application and giving the user access to the file and application via thin client is novel and is an *unobvious difference* under § 103 because it results in improved and unexpected results.

Example: _____.

URL Is Not Metadata. From the above definition, it can be seen that metadata is data about a file and is normally kept in a separate location. Neither criteria is met by a url pointing to a HTML document.

Adding to the method of Claim 132 the additional step of associating a unique identifier (or as in the amended claim: "network location or address") with metadata identifying the computer file produces unexpected results, i.e.:

Physical Differences With Accessing An HTML Document. The claimed invention has novel physical differences compared with activating a URL and accessing an HTML document. Specifically, accessing an HTML document downloads the HTML text to the web browser, and the web browser interprets the code and renders the document on the client PC. In the claimed invention, the document is not downloaded to or rendered by the web browser.

The Rejection of Claim 134 Is Overcome

Claim 134 Is Dependent on Claim 132, and so meets Requirements of § 102 and § 103.

Claim 134 is dependent upon, and incorporates all of the limitations of, claim 132. Therefore, since claim 132 meets the novelty and unobviousness requirements, so also does claim 134. Specific reasons for the rejection of claim 134 raised by the O.A. are discussed next.

Claim 134 was rejected because the Office Action stated that Simonoff discloses the application program is associated with the computer file after the activation of the hyperlink is activated as inherent feature of HTML file (underlining added).

A URL To An HTML File Specifies Network Location and Resource Access

Protocol. As is well known in the art, a hyperlink in the form of a URL points to a network resource, and the address prefix delineates the access protocol (http, ftp, gopher, telnet, and file, etc.) Other than this, there is no metadata intrinsically associated with the URL:

[T. Berners-Lee, rfc2396, August 1998: "The term "Uniform Resource Locator" (URL) refers to the subset of Universal Resource Identifiers that identify resources via a representation of their primary access mechanism (e.g., their network "location"), rather than identifying the resource by name or by some other attribute(s) of that resource", <http://www.ietf.org/rfc/rfc2396.txt>]

A URL's Access Protocol Is Specified Before URL Activation. A URL intrinsically contains the access protocol *before* activation of the hyperlink, whereas in the claimed invention, the application program can be selected after hyperlink activation. This is unobvious under § 103 because it results in an unobvious advantage- the selection of the application program can be modified based on information received after a hyperlink is activated, i.e., whether a user has a legal right to use a particular program. This is

impossible with a normal URL accessing a HTML document.

A URL Access Protocol Is Not An Application Program. A URL's 'access protocol' is physically different in many ways from the claimed invention's 'application program'. One difference is that there are a limited number of access protocols, i.e. http, ftp, gopher, telnet, etc. With the claimed invention, the number of application programs available to access a file is vastly greater: every application ever written that is capable of opening a file and that has a GUI. This constitutes both a physical difference under § 102 and an advantage over Simonoff under § 103. Another difference is that the access protocol depends specifically on the

The claimed invention's access protocol is essentially thin-client enabled 'terminal emulation' access to server-based applications that have GUIs. Currently there is no URL access protocol which uses GUI-terminal emulation.

Claimed Invention Has Physical Differences Over Telnet. Telnet is simply a character-based terminal emulation program. Text terminals commonly emulated include VT100, 200 & 300. It is typically used to provide user oriented command line login sessions between hosts on the Internet. Physical differences with claimed invention are that Telnet is not file-oriented. That is, you cannot telnet to a specific file; rather, you telnet to a specific host. Telnet does not give access to a remote application's GUI. The claimed invention rather uses a remote windowing protocol such as Citrix, Tarantella, X-Windows, or Windows Terminal Services.

Association Is Not Implementation. Perhaps the O.A. interpretation of the word 'associated', as in "Simonoff discloses the application program is associated with the computer file after the activation of the hyperlink is activated as inherent feature of HTML file", is the actual act of carrying out the prespecified access protocol after a URL is activated. However, as is well-known in the art, the association or linking of a file

network location and an access protocol happens at the time the URL is created, not after. It is clear that association is not the same as implementation, that is, carrying out a prespecified act.

Selecting Application Program After Hyperlink Activation Is Unobvious Under § 103. Selecting a server-based application program after URL activation is unobvious under § 103 because it results in new and unexpected capabilities over Simonoff and the inherent features of an HTML file. One such advantage is that this feature allows the application program to be varied based on information gathered after a URL is activated, such as the identity of the accessing party, which intrinsically cannot be known at the time of URL creation (and access protocol assigning).

The Rejection of Claim 135 Is Overcome

Claim 135 Is Dependent on Claim 132, and so meets Requirements of § 102 and § 103. Claim 135 is dependent upon, and incorporates all of the limitations of, claim 132. Therefore, since claim 132 meets the novelty and unobviousness requirements, so also does claim 135. Specific reasons for the rejection of claim 135 raised by the O.A. are discussed next.

First Reason. Claim 135 was rejected because the Office Action stated that Simonoff discloses multiple application programs are available to be associated with the computer file, and the associated application program is selected based on selection criteria chosen from the group comprising: (a) legal rights of the user to use the application programs as a design choice; (b)... (c)....

First Reason is Physically Unworkable. As detailed above in the remarks for Claim 134, paragraphs 1 & 2, a URL to an HTML file specifies network location and resource access protocol, and a URL's access protocol is specified *before* URL activation

(specifically at the time of URL creation). Knowledge of the identity of subsequent URL activators (users) cannot be known at the time of creation, so it is impossible to use this information to select an application program based on the legal rights of the user.

Second Reason The Office Action stated that Simonoff discloses multiple application programs are available to be associated with the computer file, and the associated application program is selected based on selection criteria chosen from the group comprising: (a)... (b) capabilities of the application programs; and (c) properties that are associated with the hyperlink as inherent feature of HTML file.

O. A. Misinterprets 'Application Program' As Defined In Specs. Regarding (b) capabilities of the application programs, the O.A. misinterprets the words 'application program' as clearly defined in the specifications. As detailed above in the remarks for Claim 134, paragraph 3, a URL access protocol is not an application program.

Regarding (c) properties that are associated with the hyperlink as inherent feature of HTML file, as detailed above in the remarks for Claim 13, paragraph 3, there are no properties associated with a hyperlink to an HTML file. A URL contains only the access protocol itself and the network address of the resource.

The Rejection of Claim 141 Is Overcome

Claim 141 Is A Machine Claim Containing Limitations Similar to Claim 132, and so meets Requirements of § 102 and § 103. For the reasons detailed above for claim 132, claim 141 meets the novelty and unobviousness requirements of the applicable statutes. Specific reasons for the rejection of claim 141 raised by the O.A. are discussed next.

First Reason. Claim 141 was rejected for reasons similar to Claim 132 except for the difference

of Claim 141 subpara (f) and (g). Applicant respectfully refers the Examiner to the arguments for claim 132, above. Applicant believes that these arguments adequately answer the reasons for the rejection of claim 141 that are similar to claim 132.

Second Reason. The O.A. stated that Simonoff discloses (f) a listener component capable of detecting an activation of the hyperlink over the computer network by the remote computer [Simonoff, filtering or filtered, col 10 lines 41-55, 56- col 11 line 6].

Simonoff "filtering" Is Not Hyperlink Detection: . The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

Third Reason. The O.A. stated that Simonoff discloses (g) an initiator component that, upon detection of the hyperlink activation, opens the computer file in said application program and transmits the graphical user interface of said application program to the thin client operating on the remote computer, whereby control and protection of the computer file is retained, while giving appropriate access to recipients via said server-based applications delivered thin client [Simonoff, thin client, col 2 lines 45-62; intranet and security reasons, col 8 lines 1-15; server based applications, col 15 line 55-col 16 line 11].

Simonoff "thin client" Generates GUI On Client, Not Server. As detailed above under general remarks, Simonoff's thin client, which generates and operates the user's GUI on the client PC, is not an equivalent of the claimed invention, where the user's GUI is generated by a remote application and transmitted to the thin client in essentially a terminal emulation mode.

Simonoff Sends File Data To Client, Opposite To Claimed Invention. The Simonoff citation "intranet and security reasons" is a reference to achieving data security as a feature of the network, not as a feature of the invention, by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. Simonoff teaches that actual file data (not screen refresh data, as in the claimed invention) is downloaded to the client [Simonoff, "an application...can generate data which will eventually arrive at the Universal Client device on the client host in the form of an incoming GUIScript message", col 12, lines 42-43]. In contrast, with the claimed invention security is intrinsic and does not depend on the network, because the data is retained on the server and is never downloaded, even in part, to the client. Simonoff does not retain application file data on the server as claimed.

Simonoff "server based applications" Do Not Operate GUIs. Simonoff teaches that the GUI is created on the client PC by the Universal Client applet. Thus he does not mention any GUI intrinsic to the networked (server-based) application [Simonoff, method for creating user front end GUIs for networked applications, col 5 lines 16-36]. This functions oppositely to the claimed invention where the server-based application must have a GUI which is transmitted to the user. Claim 141 has been rewritten to make clear that the remote application must have a GUI.

The Rejection of Claim 142 Is Overcome

Claim 142 Is Dependent on Claim 141, and so meets Requirements of § 102 and § 103. The machine of Claim 142 is dependent upon, and incorporates all of the limitations of, claim 141. Therefore, since claim 141 meets the novelty and unobviousness requirements, so also does claim 142. Specific reasons for the rejection of claim 142 raised by the O.A. are discussed next.

Claims 142 was rejected because the Office Action stated that Simonoff discloses the at least one processing unit is selected from the group consisting of (a) a single server; (b) a personal

computer; and (c) multiple separate computers operating as a single, logical server {Simonoff, desktop computer, LAN/WAN, Internet, col 7 lines 56-col 8 line 15}.

Claim 142 is simply a reification of the term 'at least one processing unit' from claim 141. In view of the arguments presented above that claim 141, as rewritten, is valid, applicant respectfully submits that claim 142 is now in a position of allowance.

The Rejection of Claim 151 Is Overcome

First Reason. Claim 151 was rejected for reasons similar to claim 132. Applicant respectfully refers the Examiner to the arguments for claim 132, above. Applicant believes that these arguments adequately answer the reasons for the rejection of claim 151 that are similar to claim 132.

Second Reason. Claim 151 was also rejected because the Office Action stated that Simonoff discloses:

(f) providing thin client means on said at least one remote recipient user computer adapted to receive and display a graphical user interface of said file-compatible server-based application [Simonoff, GUI script, col 7 lines 16-55; col 8 line 51-col 9 et seq.; a format compatible, col 9 lines 10-29].

Thin Client Defined Differently. The term "thin client" has different meanings under Simonoff, and in the current invention as defined in the specifications. Under the meaning as claimed, it constitutes a physical difference over Simonoff which results in new and unexpected results, as detailed above for claim 132.

Claim 151 (f) has been rewritten by replacing "a graphical user interface" with the words "a graphical user interface in a terminal emulation mode" to make clear that the claim is

for sending the GUI from the server to the client in a terminal emulation mode.

Physical Difference: Simonoff "format compatible" Is For Client, Not Server. The Simonoff citation "a format compatible" does not refer to a remotely-operating *application compatible with a computer file*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23].

Nowhere in Simonoff is the word "compatible" associated with a remote data file, because the Simonoff invention is application-oriented, not data file-oriented.

Third Reason. Claim 151 was also rejected because the Office Action stated that Simonoff discloses:

(g) providing means on said computer server system adapted to select and start said file-compatible server-based computer application in a user session [Simonoff, loads, initializes and runs, col 8 lines 26-50; filtering or filtered, col 10 lines 41-55, 56-col 11 line 6]

Simonoff "loads initializes and runs" Happens On Client, Not Server. The Simonoff citation "loads, initializes and runs" does not refer to an application program operating on a server. Rather, it refers to the Universal Client applet *running on a user's PC* [Simonoff, "...the Universal Client device is to be downloaded to the user's computer...after the Universal Client device loads, it initializes and runs" col 8 lines 47-50].

Simonoff "filtering" Is Not Hyperlink Detection: . The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the

server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

Fourth Reason. Claim 151 was also rejected because the Office Action stated that Simonoff discloses:

(h) providing file means on said computer server system which opens said computer file in said file-compatible server-based computer application [Simonoff, a format compatible, col 9 lines 10-29];

Simonoff "format compatible" Refers to Application Running on Client, Not Server. The Simonoff citation "a format compatible" does not refer to a remotely-operating *application compatible with a computer file*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23].

Nowhere in Simonoff is the word "compatible" associated with a remote data file, because the Simonoff invention is application-oriented, not data file-oriented.

Fifth Reason. Claim 151 was also rejected because the Office Action stated that Simonoff discloses:

(i) providing means on said computer server system for transmitting the human interface of said file-compatible server-based computer application to the at least one remote recipient user computer and to receive inputs to said interface from the at least one hyperlink recipient user [Simonoff, read, parse and process the reference in HTML code, col 10 lines 12-27].

Thin Client Defined Differently. The term "thin client" has different meanings under Simonoff, and in the current invention as defined in the specifications. Under the meaning as claimed, it constitutes a physical difference over Simonoff which results in

new and unexpected results, as detailed above for claim 132.

Claim 151 (i) has been rewritten by replacing "transmitting the human interface" with "transmitting the graphical user interface in a terminal emulation mode" to make clear that the claim is for sending the GUI from the server to the client in a terminal emulation mode.

The Rejection of Claim 187 Is Overcome

The method of Claim 187 Is Dependent on Claim 151, and so meets Requirements of § 102 and § 103. Claim 187 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements, as demonstrated above, so also does claim 187. Specific reasons for the rejection of claim 187 raised by the O.A. are discussed next.

Claim 187 was rejected because the Office Action stated that Simonoff discloses the at least one processing unit is selected from the group consisting of (a) a single server; (b) a personal computer; and (c) multiple separate computers operating as a single, logical server {Simonoff, desktop computer, LAN/WAN, Internet, col 7 lines 56-col 8 line 15}.

Claim 187 is simply a reification of the term 'computer server system' from claim 151. In view of the arguments presented above that claim 151 is valid, applicant respectfully submits that claim 187 is now in a position of allowance.

The Rejection of Claim 152 Is Overcome

The method of Claim 152 Is Dependent on Claim 151, and so meets Requirements of § 102 and § 103. Claim 152 incorporates all of the limitations of claim 151. Therefore, since claim 151

meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 152. Specific reasons for the rejection of claim 152 raised by the O.A. are discussed next.

Claim 152 was rejected because the Office Action stated that Simonoff discloses providing hyperlink generation means for creating said application file hyperlink and associating the hyperlink with said at least one computer file [Simonoff, read, parse and process the reference in HTML code, col 10 lines 12-27].

The claim has been rewritten to replace the words "hyperlink generation means" with "computer executable code".

Incorrect Citation. The complete quote from Simonoff shows that the word "referenced" is simply an alternate form of the word "embedded", and is not a URL:

[Simonoff, the Universal Client device advantageously can read, parse and process the GUIScript commands embedded or referenced in the HTML code of the web page containing the Applet tag for the Universal Client device, col 10 lines 17-20].

Simonoff "the reference in HTML code" Is Not A Hyperlink. From the complete reference, it can be clearly seen that Simonoff is simply referring to to be carried out locally by the Universal Client.

The Simonoff citation does not refer to detecting activation of a hyperlink or URL. The words "embedded or referenced" refers to software commands contained in a web page, i.e. GUIScript commands, and to the client applet reading and executing the GUIscript commands to create a GUI. Therefore, since the word "reference" does not refer to a hyperlink, Simonoff cannot teach how to create such a hyperlink.

The claim has been rewritten to replace the words "hyperlink generation means" with "computer executable code".

The Rejection of Claim 153 Is Overcome

The method of claim 153 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 153 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 153. Specific reasons for the rejection of claim 153 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses providing hyperlink transmission means for transmitting said application file hyperlink to said at least one hyperlink recipient user [Simonoff, downloading of web page, col 8 lines 26-50].

Claim 153 has been rewritten to correct §112 language. It now reads:

The method of claim 151 ~~61~~, further comprising the step of ~~providing hyperlink transmission means for~~ transmitting said application file hyperlink to said at least one hyperlink recipient user;

According to the O.A. citation, a web page is downloaded in response to activating a URL. However, the cited text does not address how the URL is transmitted to the user.

The arguments mentioned above under the remarks for claim 151 demonstrate clearly that claim 151 is novel under §102 and unobvious under §103. The additional step of URL transmission (claim 153) is also

The Rejections of Claims 154-156 Are Overcome

The method of claims 154-156 are dependent on claim 151, and so meet the requirements of § 102 and § 103. Claims 154-156 incorporate all of the limitations of claim 151. Therefore, since

claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 154-156. Specific reasons for the rejection of claim 154-156 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses said communications network comprises an Internet; a local area network; a uniform resource locator [Simonoff, LAN/WAN, Internet/Intranet, reference HTML code, col 8 lines 1-15; col 10 lines 13-27].

Claims 154-156 are simply reifications of the terms 'communications network' and 'hyperlink' from claim 151. In view of the arguments that claim 151 is valid, applicant respectfully submits that claims 154-156 are now in a position of allowance.

The Rejection of Claim 157 Is Overcome

The method of claim 157 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 157 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 157. Specific reasons for the rejection of claim 157 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses (a) disassociating said application file hyperlink from said at least one computer file; and (b) associating said application file hyperlink with at least one different computer file [Simonoff, embed or remove, col 15 lines 55-67].

Simonoff never mentions a hyperlink associated with a file and a remote application, as detailed above in the arguments for claim 132. Therefore, it would not be logical for Simonoff to teach dissociating said hyperlink from a specific computer file and reassociating it with a different file.

Misunderstood reference. The reference, 'embed or remove' does not refer to a hyperlink, or to dissasociating and reassociating a hyperlink. The full citation reads, 'the Universal Client...provides the mechanism to remove requirements for specific embedded display capabilities from any distributed system architecture'. This refers to the ability of the Universal Client to create a GUI on the client PC.

However, contrary to this citation, elsewhere Simonoff acknowledges that the GUI is not really independent of embedded display capabilities because it requires and uses a graphical library resident on the client PC [Simonoff, GUIScript can instantiate any GUI object common between Microsoft Windows, X-Windows, and the Java awl graphics library, col 11 lines 41-43]. In contrast, in the claimed invention, the client PC has no requirement to have any graphics library at all, since the GUI is generated on the server and transmitted to the client in essentially a terminal emulation mode.

The Rejection of Claim 158 Is Overcome

The method of claim 158 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 158 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 158. Specific reasons for the rejection of claim 158 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses modifying the application interface capability function settings of said file-compatible server-based computer application [Simonoff, program can be modified responsive to a specific scripting language, col 4 lines 34-38].

This citation is one of 21 very general objects of the invention mentioned in Simonoff. How to achieve many of the objects, including this one, is not actually disclosed in the patent.

Physical Difference. In any case, Simonoff does not disclose associating a hyperlink with a specific file and server-based application, as detailed above in the arguments for claims 151 and 132. Therefore, Simonoff does not disclose modifying the interface of said server-based application. It is the combination of said hyperlink with the modification of the server-based application's interface which is novel.

Valuable, new, improved, and unexpected results. The combination of said hyperlink with the modification of the server-based application's interface results in new and unexpected results, as detailed in the specifications.

For example, varying the server-based application's interface can improve the sender's objectives and the receiver's user experience. If the sender has the objective of allowing a viewer to only view and not modify a file, then menu items associated with modifying the file can beneficially be removed. By simplifying the application interface (removing menu items, buttons, etc.), a user can have an easier time of it. For example, if AutoCAD is the server-based application, it is well-known that AutoCAD has an extremely complex interface to create and alter CAD files. However, a viewer who is not going to modify the file would only have need of viewing and rotating tools.

Because legacy applications have not been designed as communications tools, modifying the GUIs was not contemplated by the designers. This is the unexpected part. The claimed invention makes every legacy application with a GUI into a means of communication.

The Rejection of Claim 159 Is Overcome

The method of claim 159 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 159 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated

above for claims 151 and 132, so also does claim 159. Specific reasons for the rejection of claim 159 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses the recipient user computer is of a type selected from the group consisting of: (a) a personal computer; (b) a personal digital assistant; (c) a web or internet appliance device; and (d) a television set-top box as inherent feature of web applications [see Cleron reference].

Claim 159 is simply an explanation and reification of claim 151's term 'recipient user computer', as detailed in the specifications. Therefore, since claim 151 is novel and unobvious under USC 102, so also is claim 159.

The Rejection of Claim 160 Is Overcome

The method of claim 160 is dependent on claim 152, and so meets requirements of § 102 and § 103. Claim 160 incorporates all of the limitations of claim 152. Therefore, since claim 152 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 160. Specific reasons for the rejection of claim 160 raised by the O.A. are discussed next.

The Office Action stated that Simonoff discloses the graphical user interface of said hyperlink generation means comprises a web page form [Simonoff, form, format, col 2 lines 11-25; col 11 lines 24-40; col 12 lines 1-67].

As discussed under claim 152, Simonoff does not disclose associating a hyperlink with a specific file and server-based application, as detailed above in the arguments for claims 151, 132, and 152. Therefore, Simonoff does not disclose a means to create such a hyperlink. It is the combination of said hyperlink with a step to create said hyperlink which is novel.

The citation 'form, format' refers to a web form. Claim 160 is simply an explanation and reification of claim 152's term 'graphical user interface of said hyperlink generation means', as detailed in the specifications.

The Rejection of Claim 161 Is Overcome

The method of claim 161 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 161 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 161. Specific reasons for the rejection of claim 161 raised by the O.A. are discussed next.

Claim 161 has been rewritten to divide it into two dependent claims. The first, claim 161, claims file access restriction means, and the second, claim 162, claims variable file access restriction means.

The O.A. stated that Simonoff discloses providing file access and manipulation restriction means whereby said recipient user's access to and manipulation of said at least one computer file is variable and can be set for specific sessions according to specified parameters [Simonoff, intranet and security reasons, col 8 lines 1-15].

Physical difference. As discussed above under claim 141, the Simonoff citation "intranet and security reasons" is a reference to achieving data security as a feature of the network, not as a feature of the invention, by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. Nowhere does Simonoff discuss file access restrictions. Because Simonoff does not contemplate using server-based

applications as a means of hyperlink-enabled broad communication, he does not disclose ways to restrict or modify such communication.

Valuable, new, improved, and unexpected results. The combination of said hyperlink with file access and restriction means results in new and unexpected results, as detailed in the specifications.

For example, applicable application functionality restrictions may include the following: read/write, read-only, no printing, no editing, no copying, or various other restriction on application functions as applied to the particular file. The restrictions may also be event or time-based, e.g., there may be a restriction that a user may access the file a certain number of times, view until a particular date. The access may also be based on the access location of a remote user, e.g. a user may only access the file from a certain IP address or modem phone number, or conversely, may NOT access the file from certain IP addresses or phone numbers. This may prevent unauthorized access when authentication information is compromised, or during a duress situation, or to prevent applications from being used in countries where export laws prohibit use.

The Rejection of Claim 162 Is Overcome

The Office Action stated that Simonoff discloses the file restriction means comprises a partially disabled file-compatible server-based application [Simonoff, intranet and security reasons, col 8 lines 1-15].

The method of claim 162 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 162 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 162. Specific reasons for the rejection of claim 162 raised by the O.A. are discussed next.

Physical difference. As discussed above under claim 161, the Simonoff citation "intranet and security reasons" is a reference to achieving data security as a feature of the network, not as a feature of the invention, by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. The reference does not refer to a partially disabled server-based application.

Valuable, new, improved, and unexpected results. The combination of said hyperlink where the file access and manipulation restriction means are comprised of partially disabled server-based applications results in new and unexpected results, as detailed in the specifications. For example, when accessing files, users click on a hyperlink to open the document in the proper application. Each user may be granted a security clearance level, for example, based upon rank, position, department, or other criteria. In addition to the user security level, each file preferably will also have a security level. The combination of the file security level and user security clearance level determine which files are available and the user's level of access. For example the user may or may not be able to edit, save, save as, print, copy, or the like, depending on the granted access level.

As another example, application licensing may impact application functionality. Various application versions may have different functionality for legal reasons. That is, licensing terms may require differing levels of functionality for different payments to the vendor or different circumstances. A "demonstration" license might require disabled functionality. In certain situations, AppLink recipients may actually pay for and/or subscribe to certain functions or subsets of functions, eliminating those that they would not use from the interface and simplifying their experience of the application.

The Rejection of Claim 163 Is Overcome

The Office Action stated that Simonoff discloses said file restriction means comprises a partially disabled thin-client application means [Simonoff, intranet and security reasons, col 8 lines 1-15].

The method of claim 163 is dependent on claim 161 which is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 163 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 163. Specific reasons for the rejection of claim 163 raised by the O.A. are discussed next.

Physical difference. As discussed above under claim 161, the Simonoff citation "intranet and security reasons" is a reference to achieving data security as a feature of the network, by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. The reference does not refer to a partially disabled thin-client.

Valuable, new, improved, and unexpected results. The same results as detailed above for claim 162 are also applicable here.

The Rejection of Claim 171 Is Overcome

The Office Action stated that Simonoff discloses (a) providing executable code on said computer server system for producing a computer visual interface desktop work area, (b) providing access to or links to zero or more additional applications on the visual interface desktop, (d) providing access to or links to said at least one computer file or document on the visual interface desktop; and wherein said thin client means on said at least one remote recipient computer are adapted to receive and display said computer visual interface desktop work area [Simonoff, a format compatible, col 9 lines 10-29].

The method of claim 171 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 171 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 171. Specific reasons for the rejection of claim 171 raised by the O.A. are discussed next.

Physical difference. As discussed above under claim 161, the Simonoff citation "a format compatible" refers to an application running on the client PC, not the server. The citation does not refer to a remotely-operating *computer visual interface desktop work area*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23]. Nowhere in Simonoff is the word "compatible" associated with a remote server-based application.

Valuable, new, improved, and unexpected results. This embodiment of sending a file or files within a GUI desktop makes possible links that would otherwise be impracticable. For example, with some programs, such as Adobe Photoshop™ or the Linux GIMP image editing program, the program works with multiple windows which can be minimized, moved, and resized, and which serve various specialized functions, for example as tool and color palettes. Multiple windows cannot readily be web-enabled by existing web-enabling applications. Also, those web-enabling applications have some limitations on window sizes. For example, the Tarantella™ web-enabling application does not have infinitely resizable application windows for Microsoft Windows applications; it is limited to fixed, preset application window sizes (such as 800x600) as set by the Tarantella administrator. However, if Tarantella is used to web-enable a GUI desktop, then within the desktop any application windows are infinitely resizable. Also, all the advantages of a GUI desktop then become available to an AppLink recipient. For

example, the Linux Gnome or KDE desktop has “virtual desktops,” which extend the working area of the user’s screen up to 32-fold. An application that is web-enabled directly, rather than within a GUI desktop, will typically have a working area limited to the application window size.

The Rejection of Claim 177 Is Overcome

The Office Action stated that Simonoff discloses (a) providing executable code on said computer server system for producing a computer visual interface desktop work area, (b) providing access to or links to zero or more additional applications on the visual interface desktop, (d) providing access to or links to said at least one computer file or document on the visual interface desktop; and wherein said thin client means on said at least one remote recipient computer are adapted to receive and display said computer visual interface desktop work area [Simonoff, a format compatible, col 9 lines 10-29].

The method of claim 177 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 171 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 177. Specific reasons for the rejection of claim 177 raised by the O.A. are discussed next.

Physical difference. As discussed above under claim 161, the Simonoff citation “a format compatible” refers to an application running on the client PC, not the server. The citation does not refer to a remotely-operating *computer visual interface desktop work area*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user’s PC. [Simonoff, “the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript”, col 9 lines 21-23].

Nowhere in Simonoff is the word "compatible" associated with a remote server-based application.

Valuable, new, improved, and unexpected results. A single AppLink may be hyperlink to a number of different files, such as a group of files relevant to a project. Suitable ways of implementing a multiple file AppLink include but are not limited to the following: A multi-file AppLink could simply bring the recipient to a web page that then has multiple links, one for each file. The recipient would then click on each individual link to view each file. Alternatively, a multi-file AppLink would open each file in its own application window all at once when it is activated.

The Rejection of Claim 180 Is Overcome

The Office Action stated that Simonoff discloses providing means for at least one additional user using at least one additional remote user computing device to view and optionally interact with the interface of said file-compatible server-based computer application, whereby the additional user and the first hyperlink recipient user may engage in simultaneous collaboration over said file [Simonoff, chat room, col 14 lines 1-12].

The method of claim 180 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 180 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 180. Specific reasons for the rejection of claim 180 raised by the O.A. are discussed next.

Physical difference. Simonoff discloses multiple clients sharing application GUIs, such as a chat room [Simonoff, several clients... can communicate among themselves using, in an exemplar case, the chat room paradigm, col 14 lines 7-9]. However, what is claimed

is simultaneous collaboration over a file, in essence application sharing, *in combination with* the claimed application file hyperlink, which is novel.

Valuable, new, improved, and unexpected results. Because the application file hyperlink allows links to server-based applications and files to be easily pasted into publicly-available web pages, this capability would allow the hyperlink originator the capability of instant collaboration with one or more anonymous persons who happen to have activated the hyperlink. Such a capability was not a part of the art at the time of invention. It is the ability to seamlessly combine asynchronous communication (where both parties are not required to be online at the same time, like email) with synchronous communication (like instant messaging, or application sharing, where both persons must be online at the same time) which is new, improved and unexpected.

The Rejection of Claim 181 Is Overcome

The Office Action stated that Simonoff discloses providing subscription means associated with said service whereby a remote hyperlink recipient user who is not a subscriber of said entity is solicited to subscribe to said service, as a design choice.

The method of claim 181 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 181 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 181. Specific reasons for the rejection of claim 181 raised by the O.A. are discussed next.

Physical difference. Claim 151 is physically novel over Simonoff, as detailed in the remarks for that claim. Therefore, using the invention of claim 151 as a business method is also novel.

Valuable, new, improved, and unexpected results. According to the present invention, an AppLink may be implemented as a viral marketing mechanism for service providers as follows: A link or other mechanism may be included in some or all sent AppLinks, depending for example on the subscription level of the sender or creator, whereby an AppLink recipient is advised or solicited regarding signing up with the service provider that is providing the AppLink capability to enable the recipient to send their own AppLinks. In this manner, incentive is provided to the recipient to sign up or register. According to this embodiment of the present invention, the novel capability of AppLink to deliver server-based applications and files to a broad range of anonymous or random recipients allows AppLink to be used as a viral marketing mechanism. Viral marketing typically may be thought to require two components: (1) a user motivated to use a service to communicate with others that are not currently members of the service, and (2) provision for nonmembers that are communicated with to sign up for the service. Since an AppLink is essentially a new and useful way to communicate with any number of recipients, it intrinsically provides the first component. The second condition can be satisfied by including a registration form or a link included in a recipient's introductory web page, for sent AppLinks which, if activated by the AppLink recipient, brings an AppLink recipient to an informational web page which would include that ability to register with the service, perhaps through a limited time offer.

Another example: Another embodiment where the use of AppLinks can act as a viral marketing mechanism for *Software Vendors or publishers* is by way of an AppLink Recipient Demo Application License. According to this embodiment, a free "demo" license for AppLink recipients allows software vendors to present their application to potential customers via the viral marketing mechanism of AppLink.

The Rejection of Claim 182 Is Overcome

The Office Action stated that Simonoff discloses a log of accesses to said computer file is kept on said computer server system [Simonoff, database, col 2 lines 46-62].

The method of claim 182 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 182 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 182.

Specific reasons for the rejection of claim 182 raised by the O.A. are discussed next.

Physical difference. The Simonoff citation refers to a Java applet accessing a server-based database. Simonoff says nothing in the cited reference or elsewhere about *creating* a database of *file access items*, which is what is claimed.

Valuable, new, improved, and unexpected results. According to the present invention, settable AppLink properties pertaining to access tracking may include, but are not limited to, recordation and/or notification of: each individual access or given number of accesses; the ID data of each accessor; the time of access of one recipient or the consolidated or average access time of a group of recipients; the AppLink session time; the file access and/or manipulation details; or the interaction playback of a recipient session.

The extreme detail of recorded access far exceeds a simple email receipt. For example, if a data file is sent as an email attachment, no information about its opening or use is available. In contrast, in the claimed invention, because the file remains on the server and the application runs there also, every detail of its access and manipulation can be

recorded (up to and including actual session playback) for tracking or any other purpose. These are valuable, new, improved, and unexpected results flowing from the act of recording access details.

The Rejection of Claim 188 Is Overcome

The Office Action stated that Simonoff discloses the hyperlink activation is detected on a detection computer separate from the remote computer [Simonoff, the browser identifies the device to be downloaded to the user's computer, col 8 lines 26-50].

Correction of Improper Antecedent. Claim 188 had an improper antecedent (remote computer) basis in parent independent claim 187. It is corrected to read:

The method of claim 187, wherein said multiple, separate computers operating as a single, logical server comprise _____
the computer software executable code which detects the execution of the hyperlink by said hyperlink recipient user over said communications network operates on a computing device physically separate from separate from the remainder of said server system but connected through said communications network.

The method of claim 188 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 182 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 188. Specific reasons for the rejection of claim 188 raised by the O.A. are discussed next.

Physical difference. The citation refers to a web page containing an applet tag which causes the universal client applet to be downloaded to the client PC. The server detects the activation of the applet tag in the web page. In contrast, in the claimed invention, the

server detects the activation of a URL associated not with an applet but with a specific application file and a compatible server-based application.

Valuable, new, improved, and unexpected results. Because most PCs do not have a web server capability installed, separating out this capability into the server system, and having the server system utilize the PC as a file storage and application server would allow AppLink capability to easily be extended to ordinary PCs with minimal modification. That is, said server system would incorporate PCs connected to the network as the part of the server system that functions as file and/or application servers. This results in whatever applications a user has installed on his PC (however esoteric) being available for AppLink usage. Meaning that the rest of the server need not have numerous esoteric and rarely used applications installed on itself.

The present invention may be implemented via an AppLink creator's own Internet-connected computer, with the creator's machine performing the functions of an AppLink Server and Application Server. In order for this implementation to be transparent to the recipient, this computer will preferably remain connected to the network at all times. Accordingly, ISPs and website hosting services, as well as entities or individuals maintaining their own internet servers, may in many cases implement this capability almost immediately. In the situation in which an AppLink incorporates a file not resident on the AppLink Server, e.g., the file is stored on a different Network-connected computer or data storage device than the AppLink Server, the AppLink Server preferably will record the location of the file at the time the AppLink is created, and will use a system to open it from that location, or it will upload a copy of the file to the AppLink Server. An Application Server must be able to access the file being AppLinked in order to open and run it within an appropriate server-based application.

The Rejection of Claim 189 Is Overcome

The O.A. states that Simonoff discloses the application program and the computer file are stored on a storage device not directly connected to the remote computer as a design choice. The O.A. implies that this would be obvious to a person with ordinary skill in the art at the time of invention.

Claim rewritten and divided for clarity. Claim 189 is rewritten into two separate claims for clarity, and to correct an improper antecedent in claim 151, the claim they are dependent upon, i.e., "the remote computer" which should read as "multiple, separate computers operating as a single, logical server". They now read:

Claim 189. (old claim 189) The method of claim 151, wherein one of said multiple, separate computers operating as a single, logical server comprise a separate computer operating as an application server.

Claim 189a (NEW) (old claim 189) The method of claim 151, wherein one of said multiple, separate computers operating as a single, logical server comprise a separate computer operating as a file server.

Physical Difference. Claims 189 and 189a incorporate the application file hyperlink of claim 151 and so are novel as detailed in the comments for claims 151 and 132. Physical differences are similar to those mentioned for claim 188.

Valuable, new, improved, and unexpected results. Because most PCs do not have a web server capability installed, using a PC as a file storage and application server would allow AppLink capability to easily be extended to ordinary PCs with minimal modification, and with the creator's PC performing the functions of an application server and/or file server. In order for this implementation to be transparent to the recipient, this computer will preferably remain connected to the network at all times. Accordingly,

ISPs and website hosting services, as well as entities or individuals maintaining their own internet servers, may in many cases implement this capability almost immediately.

The Rejection of Claim 190 Is Overcome

Claim 190 is withdrawn.

The Rejection of Claim 191 Is Overcome

The Office Action stated that Simonoff discloses downloading the thin client from the server system to said at least one remote recipient user computing device [Simonoff, thin client, col 2 line 47].

The method of claim 191 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 191 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 191.

Specific reasons for the rejection of claim 191 raised by the O.A. are discussed next.

Physical difference. The Simonoff citation refers to a thin client which is physically different from a thin client as claimed and defined in the specifications. Please see remarks for claims 132 and 151 for details.

Valuable, new, improved, and unexpected results. Downloading the thin client to the recipient's PC allows the claimed invention to give file and application access via hyperlink to virtually any internet-connected computing device. Because the present invention's intent is to allow anonymous, third-party access to remote applications and

files, it incorporates steps to assure compatibility of a recipient's PC to receive and display the remote application and file, i.e., to ensure that a compatible thin client is present on the recipient's PC, by downloading one if one is not already present.

This results in the many advantages of said application file hyperlink, as outlined in the comments for claim 151 and 132, being made available to most if not all personal computing devices, and thus extending its utility as a communications method.

The Rejection of Claim 192 Is Overcome

The Office Action stated that Simonoff discloses downloading the thin client occurs only after said at least one remote recipient user computing device is examined and found not to already have the thin client available as a design choice. The O.A. implies that this would be obvious to a person with ordinary skill in the art at the time of invention.

Correction of claim number that claim 192 is dependent upon. Claim 192 is corrected to be dependent on the immediately previous claim, or claim 191.

The method of claim 192 is dependent on claim 191, and so meets requirements of § 102 and § 103. Claim 192 incorporates all of the limitations of claim 191. Therefore, since claim 191 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 192.

Specific reasons for the rejection of claim 192 raised by the O.A. are discussed next.

Physical difference. Claim 192 is essentially a narrowing of claim 191, which is physically different from Simonoff, in that the Simonoff citation refers to a thin client which is physically different from a thin client as claimed and defined in the specifications. Please see remarks for claims 132 and 151 for details.

Valuable, new, improved, and unexpected results. Downloading the thin client to the recipient's PC allows the claimed invention to give file and application access via hyperlink to virtually any internet-connected computing device. Because the present invention's intent is to allow anonymous, third-party access (among others) to remote applications and files, it incorporates steps to assure compatibility of a recipient's PC to receive and display the remote application and file, i.e., to ensure that a compatible thin client is present on the recipient's PC, by downloading one if one is not already present.

This results in the many advantages of said application file hyperlink as outlined in the comments for claim 151 and 132, being made available to most if not all personal computing devices, and thus extending its utility as a communications method.

The Rejection of Claim 193 Is Overcome

The Office Action stated that Simonoff discloses opening the computer file occurs only after a determination is made that the hyperlink associated with the computer file is still active [Simonoff, provide heart beats to device, col 16 lines 45-61].

Claim rewritten. Claim 193 is rewritten to change "is still active" to "is still active or enabled".

The method of claim 193 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 193 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 193.

Specific reasons for the rejection of claim 193 raised by the O.A. are discussed next.

Physical difference. The Simonoff citation refers to determining whether a remote device is still connected to the server *during* an application session. The claimed

invention, in contrast, has the server system determining whether a hyperlink is still active *before* initiating a session with a remote device.

Valuable, new, improved, and unexpected results. This novel physical structure is unobvious because the invention as claimed allows protection of intellectual property (the computer file) by allowing a hyperlink creator to disable it at any time, thus cutting off access to the file. This valuable, unexpected result is in contrast to the alternative of an email attachment, which, when sent, is irretreable, and in contrast to Simonoff, who does not address the issue of I.P. protection at all, except indirectly through a secure network [Simonoff, the computer sustem 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12].

The Rejection of Claim 194 Is Overcome

The Office Action stated that Simonoff discloses changes are made to the computer file after the hyperlink is created, and the hyperlink remains associated with the latest version of the computer file. No citation is given.

The method of claim 194 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 194 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 194.

Specific reasons for the rejection of claim 194 raised by the O.A. are discussed next.

Physical difference. No Simonoff citation is given by the O.A. However, as detailed in the remarks for claim 132 and 151, Simonoff does not disclose a hyperlink associated with a specific file and server-based application. Therefore, he does not detail what happens to such a file after creation of such a hyperlink.

Valuable, new, improved, and unexpected results. Changing the file after hyperlink creation is unobvious because it results in the hyperlink being associated with the most current copy of a file, unlike an email file attachment, which cannot be updated. This capability also allows *unexpected results* like allowing a recipient to bookmark a particular application file hyperlink, said hyperlink for example always being connected to the latest issue of a subscription magazine. This is unobvious over a simple HTML file for the advantages detailed in the remarks for claims 132 and 151.

The Rejection of Claim 195 Is Overcome

The Office Action stated that Simonoff discloses the application program is not operating on the remote computer until after the activation of the hyperlink is detected [Simonoff, changing the GUI script, col 4 lines 38-46].

The method of claim 195 is dependent on claim 151, and so meets requirements of § 102 and § 103. Claim 195 incorporates all of the limitations of claim 151. Therefore, since claim 151 meets the novelty and unobviousness requirements of the applicable USC, as demonstrated above, so also does claim 195.

Specific reasons for the rejection of claim 195 raised by the O.A. are discussed next.

Physical difference. The citation refers to changing the GUI (not the application program) over long periods of time. [Simonoff, the same software... can be used repeatedly over several generations; changing the GUIscript changes the GUI presented to the operator]. It does not address the time when a particular application begins operating.

Valuable, new, improved, and unexpected results. Starting the application program after the hyperlink is activated is unobvious because it allows the choice of the application program to be made using data not available until after the hyperlink is

activated, such as data about the identity of the hyperlink activator (recipient), or data about what applications the recipient is legally allowed to use. *This valuable result is not addressed or even contemplated by Simonoff.*

The Claims Rejections Under 35 USC § 103 (a)

The Rejection of Claim 172 Is Overcome

The Office Action stated that Simonoff discloses (a) detecting said hyperlink activation and (c) opening the file copy in said file-compatible server-based application instead of the original computer file [Simonoff, browser identifies the Universal Client device to be downloaded, col 8 lines 26-50].

Reason (a) for rejection. The Simonoff citation does not refer to detecting activation of a hyperlink or URL, thus clearing the first reason (a) for rejection. However, in any case, details of why Simonoff does not disclose (a) is discussed in more detail in the discussion for claim 132.

Reason (c) for rejection. Misunderstood Reference. The Simonoff citation refers to the Universal Client applet being downloaded to and running on a user's PC [Simonoff, "...the Universal Client device is to be downloaded to the user's computer...after the Universal Client device loads, it initializes and runs" col 8 lines 47-50]. **Simonoff "loads initializes and runs" Happens On Client, Not Server.** "loads, initializes and runs" does not refer to an application program operating on a server. Rather, it refers to the Universal Client applet *running on a user's PC*. This is a major physical difference between Simonoff and the present invention.

The O.A. stated that "However Simonoff does not explicitly detail assigning a **guest account** to said recipient user; and (b) creating a copy of said at least one computer file in the guest account.

It was well known in the art that the guest/temporary account was created for the short time user on network such as Internet user."

Account types physically different from known art. The O.A assertion that guest/temp accounts were well known in the art is not correct. There is no suggestion that an account *of the type needed for running an application with a GUI*, such as Microsoft Word, be used as a guest account or be created prior to use for temporary usage. This is because computer system accounts of this type were known in the art to be used only with named user usage. There was no suggestion that multiple unnamed users would access a given desktop-type application user account, because to give multiple users access to each user's private files and applications would violate basic security practices.

Without the novel advantages of the present invention, which convert legacy applications into a method of communication and collaboration, there is no reason to create a system to give access to such application user accounts, or to manage them before and after access.

The O.A. goes on to state that "A skilled artisan would have motivation to improve the thin client access to Internet and found Beer teaching. Beer taught URL login wherein the guest account is used for the URL login {Beer, guest account, col 1 lines 29-38; java applet, col 1 lines 59-col 2 line 7; col 3 lines 48-55}. Therefore it would have been obvious to an ordinary skill in the art at the time the invention was made to incorporate the guest account as taught by Beer into Simonoff's apparatus in order to utilize the thin client. Doing so would provide the security to the thin client to access and execute Internet application locally."

Beer Teaches Oppositely. The full Beer "guest account" citation makes clear that Beer actually teaches away from using a guest account: [Beer, a user may wish to log into a network via a public terminal...In such a situation there is no way to maintain a login account for every possible person who might use the public computer, and "*guest*"

accounts have little or no real value, since they do not let a user access all of the resources to which that user has privileges."

In fact, the function of the Beer invention is to *avoid using guest accounts* and allow named users to access *their own accounts* and account resources remotely via the multiple login automation features of his invention. There is no suggestion anywhere of guest accounts being created for any purpose.

The full Beer citation containing the words "java applet" make clear that the meaning is to allow the user to access files by the mechanism of downloading the file to the local machine and opening it in a java applet downloaded to the local machine also. [Beer, ...it is possible to bring small applications called "applets" from a server to a client and execute such applications locally, col 3 lines 50-52]. This is opposite to the claimed invention where the file stays on the server and the application runs there also. Again, no mention of guest accounts in the "java applet" citation.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. teach oppositely to the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion that they be combined as the O.A. suggests.

Valuable, new, improved, and unexpected results. By creating a guest user application account on the server and copying the linked-to file into the guest account from the originator's account at the time of AppLink activation, security is obtained for the AppLink originator by not granting access to third parties to the originator's personal application account, and thus to other files or personal information not intended to be shared.

From the specifications: Accessing an AppLink with a "dummy" user account rather than

directly from the AppLink creator's account (with file access restrictions, of course), the damage which an unscrupulous, "hacker" AppLink recipient could do may be restricted. For example, there would be no way for such a recipient to access or delete other files contained in an AppLink sender's service account.

The Rejection of Claim 173 Is Overcome

The Office Action stated that Simonoff discloses the guest account is created before being assigned to the user [Beer, guest account, col 1, lines 29-38].

Claim rewritten. The claim has been rewritten to clarify what is claimed. It now reads: The method of claim 82, wherein the guest account is created ~~before being assigned to the user~~ after detecting said hyperlink activation.

Misunderstood Reference. As detailed in the remarks for claim 172, the citation teaches away from using a guest account. As a consequence, Beer does not address the question of when to create such an account.

Valuable, new, improved, and unexpected results. Creating a guest user application account on a server requires substantial time and effort. Pre-creating a series of application accounts for fictional users speeds up functional response to AppLink activation by a recipient. In the art, there has been to date no need to precreate such accounts because their use by other than named users has not been contemplated.

The Rejection of Claim 174 Is Overcome

The Office Action stated that Simonoff discloses the guest account is deleted upon the closing of the computer file by the user [Beer, guest account, col 1, lines 29-38; java applet, col 1 lines 59-col 2 line 7; col 3 lines 48-55].

Claim rewritten. The claim has been rewritten to incorporate the alternative of clearing data items from the guest account after the computer file has been closed. No new matter has been added. It now reads:

Claim 174. The method of claim 172, further comprising the steps of either

- (a) deleting said guest account upon the closing of the computer file by the user, or
- (b) deleting or cleaning data items from said guest account and returning said guest account to available status upon the closing of the computer file by the user.

Misunderstood Reference. Please see the remarks for claim 172 which addresses the Beer citations. As mentioned therein, the citations teach away from using a guest account. As a consequence, Beer does not address the question of what to do with such accounts after they are used.

The Rejection of Claim 175 Is Overcome

The Office Action stated that Simonoff disclose predefining a plurality of guest accounts, and further wherein one of the predefined guest accounts is assigned to the user after hyperlink activation [Beer, guest account, col 1, lines 29-38; java applet, col 1 lines 59-col 2 line 7; col 3 lines 48-55].

Misunderstood Reference. Please see the remarks for claim 172 which addresses the Beer citations. As mentioned therein, the citations teach away from using a guest account. As a consequence, Beer does not address the question of what to do with such accounts in combination with the claimed invention's application file hyperlinks.

The Rejection of Claim 176 Is Overcome

The Office Action stated that Simonoff disclose predefining a plurality of guest accounts, and further wherein one of the predefined guest accounts is assigned to the user after hyperlink activation [Beer, guest account, col 1, lines 29-38; java applet, col 1 lines 59-col 2 line 7; col 3 lines 48-55].

Misunderstood Reference. Please see the remarks for claim 172 which addresses the Beer citations. As mentioned therein, the citations teach away from using a guest account. As a consequence, Beer does not address the question of what to do with such accounts in combination with the claimed invention's application file hyperlinks.

The Rejections of Claims 136-140, 143-150, 165, 170, 183-186, 196-200 Are Overcome

The Office Action stated that the above claims were rejected under 35 USC § 103 as obvious over Simonoff et al [Simonoff, 6,005,568] in view of Shiigi [6,304,898 B1].

The Rejection of Claim 136

The Office Action stated that, as per claim 136, Simonoff discloses a method for handling incoming file (e-mails) at a file server (e-mail gateway) comprising:

(a) identifying a data file attachment on an incoming file (e-mail) addressed to a recipient [Simonoff, filtering or filtered, col 10 lines 41-55, 56-col 11 line 6];

Simonoff "filtering" Is Not Hyperlink Detection: . The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

(b) creating a copy of the identified attachment on a storage device accessible by the file server (e-mail gateway) [Simonoff, a format compatible, col 9 lines 10-29; read, parse and process the reference in HTML code, col 10 lines 12-27; filtering or filtered, col 10 lines 41-55, 56-col 11 line 6];

Simonoff "format compatible" Refers to Application Running on Client, Not Server. The Simonoff citation "a format compatible" does not refer to a remotely-operating *application compatible with a computer file*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23]. Nowhere in Simonoff is the word "compatible" associated with a remote data file, because the Simonoff invention is application-oriented, not data file-oriented.

Simonoff "the reference in HTML code" Is Not A Hyperlink. The Simonoff citation does not refer to detecting activation of a hyperlink or URL. The word "reference" refers to GUIScript commands, and to the client applet reading and executing the GUIscript commands embedded in a HTML document to create a GUI. **Incomplete Citation.** The

complete quote from Simonoff shows that the word "referenced" is simply an alternate form of the word "embedded", and is not a URL:

[Simonoff, the Universal Client device advantageously can read, parse and process the GUIScript commands embedded or referenced in the HTML code of the web page containing the Applet tag for the Universal Client device, col 10 lines 17-20].

Simonoff "filtering" Is Not Hyperlink Detection: . The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

(c) creating a reference or hyperlink associated with the attachment copy [Simonoff, URL, col 9 line 67; read, parse and process the reference in HTML code, col 10 lines 12-27; filtering or filtered, col 10 lines 41-55, 56-col 11 line 6];

Simonoff "URL" Doesn't Point To A File- Physical Distinction. The cited Simonoff reference "URL" refers to a hyperlink pointing to the server *host*, and not to a particular computer file [Simonoff, the GUIScript downloaded from server host 100a includes the URL pointing to server host 100n of fig. 2, col 9 lines 66-67].

See remarks, above, on claim 136, subpara (b) for comments on the citations "read, parse and process the reference in HTML code", and "filtering or filtered".

(d) deleting the attachment from the incoming file (e-mail) and adding said hyperlink to the file (e-mail) [Simonoff, URL, col 9 line 67; read, parse and process the reference in HTML code, col 10 lines 12-27];

See remarks, above, on (b) for comments on the citations "read, parse and process the reference in HTML code", and remarks on (c) for "URL".

(e) operating on a remote computer remote from the recipient an application program associated with and compatible with the attachment copy [Simonoff, a format compatible, col 9 lines 10-29];

See remarks, above, on (b) for comments on the citation "a format compatible".

(f) opening the attachment copy in the application program running on the computer remote from the recipient upon activation of the hyperlink [Simonoff, remote location, col 16 line 63];

Simonoff "remote location" Is For Data Storage, Not Operating An Application.

The Simonoff citation "remote location" refers to storing user information, such as screen size, remotely, i.e., on the server, and not to operating an application program compatible with or capable of loading the computer file. [Simonoff, , information on each user such as screen preferences... may be stored at a remote location, e.g., server host 100, col 16 lines 62-64].

(g) operating a thin client on a local computer used by the recipient, the thin client allowing the recipient to provide input to and receive output from the application program running on the remote computer, whereby the file (e-mail) attachment cannot pose a virus danger to said local computer since it is manipulated remotely, and whereas the local computer has no need of a compatible program to open said attachment {Simonoff, thin client, col 2 lines 45-62; intranet and security reasons, col 8 lines 1-15; filtering or filtered, col 10 lines 41-55, 56- col 11 line 6; server based applications, col 15 lines 55-col 16 line 11}.

Thin Client Physically Different Than Simonoff. As detailed above in the remarks for claim 132, The meaning of the term "thin client", as known in the art, refers to a simple client program or device which relies on most of the functionality of the system being

located on the server.

"A term used in the claims may be given a special meaning in the description, but no term may be given a meaning repugnant to the usual meaning of the term."
MPEP 608.01(o); 2173.05(a).

The applicant's more specific, distinct, and non-repugnant meaning of "thin client", as described in the specifications, is wherein the GUI is *generated or created* on the server and transmitted and displayed to the user via the thin client, in essentially a *terminal emulation* mode. The major processing done by the thin client is to send user input (keystrokes and mouse movements) to the server and receive the updated GUI state from the server and display it on the user's screen. Examples of thin clients of this type known in the art include Citrix and Tarantella. These major commercial products demonstrate clearly that the applicant's definition is within the meaning of the art.

Simonoff's thin client, which generates and operates the user's GUI on the client PC, is not an equivalent of the claimed invention, where the user's GUI is generated by a remote application and transmitted to the thin client in essentially a terminal emulation mode.

Simonoff Sends File Data To Client, Opposite To Claimed Invention. The Simonoff citation "intranet and security reasons" is a reference to achieving data security by using a LAN not connected to the Internet [Simonoff, the computer system 1 advantageously can be a detached local area network or intranet for practical and security reasons, col 8 lines 10-12]. Simonoff does not retain application file data on the server as claimed. In fact, Simonoff teaches that actual file data (not screen refresh data, as in the claimed invention) is downloaded to the client [Simonoff, "an application... can generate data which will eventually arrive at the Universal Client device on the client host in the form of an incoming GUIScript message", col 12, lines 42-43]. In contrast, with the claimed invention security is intrinsic and does not depend on the network, because the native

data is retained on the server and is never downloaded, even in part, to the client.

Nowhere does Simonoff discuss file access restrictions. Because Simonoff does not contemplate using server-based applications as a means of hyperlink-enabled broad communication, he does not disclose ways to restrict or modify such communication.

See remarks, above, on (b) for comments on the citation "filtering or filtered".

Simonoff "server based applications" Do Not Operate GUIs. Simonoff teaches that the GUI is created on the client PC by the Universal Client applet. Thus he does not mention any GUI which is intrinsic to, or already a part of, the networked (server-based) application [Simonoff, method for creating user front end GUIs for networked applications, col 5 lines 16-36]. This functions oppositely to the claimed invention where the server-based application must have a GUI which is transmitted to the user.

The O.A. stated, however, that Simonoff does not detail explicitly the electronic file as email attachment. A skilled artisan would have motivation to modify Simonoff's apparatus and found the Shiigi teaching. Shiigi discloses an electronic messaging system using JAVA applet (i.e.: thin client), Email server and gateway to send email with attachments [Shiigi, col 4 lines 43-63; col 5 line 34-col 6 line 67; col 7 lines 1-65].

Shiigi Patent Summary. Shiigi discloses a method and system for creating and sending graphical email. One way it does this is to download a drawing JAVA applet which allows writing to be drawn freehand and saved as an image file, such as GIF. Another way is to upload the image file to a server, which analyzes the file and interprets the drawing/writing in terms of text. This text is then forwarded as a normal email.

Misunderstood Reference: Shiigi JAVA Applet is not a thin client. The O.A. confuses java applets with thin clients. A thin client can be composed of a java applet, but a java applet does not have to be a thin client. In the case of Shiigi's disclosure, the Shiigi JAVA applet refers to a Java drawing program that either accesses a web browser's email client, or incorporates such capability in itself [Shiigi, the handwriting messaging component is a Java applet downloadable to a web page for an email client of a standard web browser,

col 2 lines 52-54]. There is no suggestion in Shiigi that the Java applet is a thin client. In fact, Shiigi does not mention the words "thin client" at all.

Shiigi Email Client Operates Locally. Shiigi explicitly teaches that the email client is downloaded to the local PC, and does not operate remotely [Shiigi, in the preferred implementation, the server computer stores the graphical email handling software that is downloaded to the client computers, col 4 lines 29-32], if it does not utilize the email client of the web browser (see above comments for Shiigi JAVA applet). A web browser email client also operates locally, of course.

The O.A. states that it would have been obvious to an ordinary skill in the art at the time the invention was made to incorporate the technique of using Email over the thin client with Java applet. Doing so would allow the web user to compose, manipulate, send and view or print handdrawn email messages.

The present invention does not claim any capabilities regarding sending handdrawn email messages.

Shiigi does not disclose or teach accessing a remote email client application over a thin client with a Java applet.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. teach oppositely to the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion that they be combined as the O.A. suggests.

The Rejection of Claim 137 Is Overcome

The Office Action stated that claim 137 contains similar limitations set forth in claim 136, and so was rejected for the same rationale set forth in claim 136.

Claim 136 has been clearly shown to be patentable under USC 103 as detailed above under the remarks for claim 136. Therefore, Applicant respectfully submits that claim 137 is in an allowable form for similar reasons.

The Rejection of Claim 138 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose imposing file access restrictions on said attachment copy [Shiigi, file attachment, col 7 lines 1-col 8 line 30]

Misunderstood Reference. The Shiigi citation "file attachment" refers to the GIF file of handwriting drawing being sent to a recipient as an email file attachment. Sending a GIF file (whether the image is of handwriting or not) as an email attachment is well known in the art. Nowhere in the cited reference is any restriction on the image file or attachment mentioned. It would not make sense to impose such a restriction, since the point of the invention is to grant access (get the image to the email's recipient), and not to restrict access to the GIF file.

The Rejections of Claim 139, 140 Are Overcome

The Office Action stated that Simonoff-Shiigi disclose setting a variable level of file access restrictions based upon a security level associated with the recipient, whereby attachment access can be varied according to the trust placed in a particular recipient individual as represented by said security level [Shiigi, authenticate, password, col 8 lines 35-67].

The Shiigi citation refers to a user sending an image of handwriting to another user in an Instant Messaging mode. Authenticating the user refers to simply confirming that the sending user is a valid (registered) user [Shiigi, when the client logs onto the server, the server looks up the user name and password in order to authenticate that person as a registered user, col 8 lines 42-45]. This is normal primary user authentication to decide whether to grant access to the server application (Instant Messaging Java Handwriting Server), which is well known in the art.

However, what is claimed is assigning a security level to a user over and above the mere verification that a user is valid. There is no suggestion in Shiigi that this be done. There would be no reason to do this, since Shiigi does not discuss file security.

The Rejection of Claim 143 Is Overcome

The Office Action stated that claim 143 contains similar limitations set forth in claim 136, and so was rejected for the same rationale set forth in claim 136.

Novelty. Claim 136 has been clearly shown to be patentable under USC 102 and 103 as detailed above under the remarks for claim 136. Therefore, Applicant respectfully submits that claim 143 is in an allowable form for similar reasons.

No suggestion to combine references. As detailed above for claim 136, all the references mentioned by the O.A. teach oppositely to the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion that they be combined as the O.A. suggests.

Valuable, new, improved, and unexpected results. The ability to create an AppLink to a computer “GUI Desktop” which is running on a remote application server allows that desktop to be used as a Web Portal Home Page, as shown in Figure 33. This “GUI Desktop” could preferably include one of the open source Gnome or KDE desktops for

Linux, since the required code is open to public access and free of charge, or one of the Microsoft Windows desktops (win 95, 98, Me, 2000, etc).

Personalized “home” pages as generated by major web portals such as Yahoo™ have evolved over time to contain various features, including personalized news, weather, stock quotes, email, an instant messenger application, calendars, and file storage . These functions, although generally well designed, are limited by the functionality of the various web page languages such as HTML and JavaScript. If an entire web page refresh is not to occur every time a viewer initiates an action, JavaScript or it’s equivalent must be written into the page itself to anticipate such an action and have instructions for generating a response. It is impossible to write into a web page with Javascript the full functionality of a standard software application. Therefore, these web page functions are limited in functionality and poorly performing compared to native software applications. Also, generally a web page is a static thing, with the content unchanging until a user initiates a web page “refresh”. With a (probably password-protected) AppLink to a computer’s “GUI Desktop” work area, all of these limitations can be overcome. Applications can be fully-fledged programs, such as the Microsoft Outlook™ email program, rather than glorified web-based forms. Current events can be streamed to the user real-time by the program without refresh. Plus the Desktop can run major applications which have no HTML equivalent. The desktop can be customized by the user to a far greater extent than can the HTML version

The Rejections of Claims 144, 146, 148 Are Overcome

The Office Action stated that Simonoff-Shiigi disclose (a) providing application hyperlink generation means on said computer server system for creating said application hyperlink; and (b) providing application hyperlink transmission means for transmitting said application hyperlink to said at least one recipient user, whereby a wide variety of potentially anonymous users are

provided simple and easy access to said server-based communications applications delivered via thin client, irrespective of the operating system and the overall capabilities of the recipient's computer, and allowing effortless access to virtually any communications application to be placed anywhere a hyperlink can be placed, such as an email or web page [Simonoff, thin client, col 2 lines 45-62; intranet and security reasons, col 8 lines 1-15; URL, col 9 line 67; read, parse and process the reference in HTML code, col 10 lines 12-27; server based applications, col 15 lines 55-col 16 line 11].

Misunderstood references. Reasons why the Simonoff citations are physically different than the claimed invention are detailed above for claim 136. Please refer to the comments for that claim.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. teach oppositely to the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion that they be combined as the O.A. suggests.

Valuable, new, improved, and unexpected results- Claim 146: The combination of said hyperlink with thin-client delivered *communications applications* results in suprising and valuable capabilities, to wit:

The Sender has a communications program (second communications means). It is not necessarily server-based. Most likely it will operate on the sender's PC. It could be any type of communications program, for example Instant Messaging. The Sender doesn't know, especially with anonymous third parties, whether those potential recipients have a compatible Instant Messaging program. However, by sending access to a server-based Instant Messaging communications program (first communications means) which IS compatible with the Sender's I.M. program, the sender can be assured that the recipient can communicate with him. Also, the server-based I.M. program can be preconfigured to

have the SENDER'S address to facilitate the use of that program to communicate with him.

In this invention, the server-based application is one or more communications programs, such as an e-mail, Instant Messaging, Chat (IRC), Voice Chat (Voice Over Internet Protocol), or Video Chat application or executable. The communications program may preferably be preconfigured with the communications address of the sender. The Communications AppLink can then be used by the AppLink recipient to communicate with the sender, who will have a compatible communications program, either server-based, or locally installed. This "CommLink" could be sent along with an AppLink to another file that the sender wishes to collaborate with the recipient about, or have the recipient view and discuss. Using AppLink, access to one user's communication tools could be sent to another user via server-based communications programs in a web page or e-mail. Suitable applications may include Instant Messaging, as in Figure 19b, email, or voice over IP and video conferencing in those instances in which the local client program can access the relevant components of the local machine, such as the microphone or WebCam. Figure23 is a depiction of a screen view for an alternate implementation of a communication link according to an embodiment of the present invention. The recipient could communicate with the sender despite having no compatible communications programs installed on his local machine. This would allow guaranteed reliable communication with any number of recipients in a manner fully supported by both/all users, with no problems with incompatible or conflicting proprietary communications applications or executables. The communications applications could be configured, according to the desires of the sender, to allow the recipient to only communicate with the sender, for example according to the sender's addresses already set up in the application.

The Rejection of Claim 145 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose a first communications means consisting of a communications program or application [Shiigi, java applet, col 4 lines 43-63; col 5 line 34-col 6 line 67; col 7 lines 1-67].

Physical difference. The Shiigi citation is for a drawing and optionally an email client java applet which is downloaded to the client PC, and operates there. In contrast, the communications means mentioned in the claim operates on the server, and is explicitly delivered via thin client. As discussed above under remarks for numerous claims, Simonoff's meaning of thin client is different from the meaning as defined in the current application's specifications as essentially a terminal emulation-type thin client.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. do not teach the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion that they be combined as the O.A. suggests.

Valuable, new, improved, and unexpected results. Please see remarks for claims 144, 146, and 148 for details of the valuable and unexpected results which also flow from claim 145.

The Rejections of Claims 147, 149, 170, 184 Are Overcome

The Office Action stated that Simonoff-Shiigi disclose said first and second communications means comprises programs selected from the group consisting of: (a) an email program (b) an instant messaging program; (c) a voice-over-internet-protocol

Misunderstood references. A description of how the Simonoff citations are physically different than the claimed invention are detailed above for claim 136. Also included for

that claim are reasons why the combination of Simonoff-Shiigi are not operative. The suggested combination is not suggested in the references.

Claim 147 is simply a reification of claim 146, and clears the requirements of USC 102 and 103 for the reasons detailed for claim 146.

Claim 149 is dependent upon a hyperlinked-to computer GUI desktop (claim 148) , and is a reification of that claim. Therefore it clears the requirements of USC 102 and 103 for the reasons detailed for claim 148.

Claim 170 is dependent on claim 153, and is simply a reification of that claim, and clears the requirements of USC 102 and 103 for the reasons detailed for claim 153.

Claim 184 is dependent on claim 183, and is simply a reification of that claim, and clears the requirements of USC 102 and 103 for the reasons detailed for claim 183.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. do not teach the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion in the references themselves that they be combined as the O.A. suggests, and the **abundant valuable, new, and unexpected results** (as detailed above for numerous claims) show that this combination is non-obvious.

The Rejection of Claim 150 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose providing access to said computer visual interface desktop to subscribers of said commercial service, whereby said application hyperlink

recipient may utilize said visual interface desktop in a manner similar to accessing a personalized home web page which utilizes HTML or javascript code, but with the full functionality of standard software applications instead of the limited functionality of said HTML or javascript code [Shiigi, col 4 lines 43-63; col 5 line 34-col 6 line 67; col 7 lines 1-65].

Physical difference. Claim 150 is the invention of claim 148 used in a commercial service. Claim 150 is dependent on claim 148, which in turn is dependent on claim 143. For the reasons detailed in those claims, claim 150 is novel and results in valuable, new, and unexpected results. These results show that this combination is non-obvious when used as a commercial service.

Neither Simonoff nor Shiigi use a thin client as the term is defined in the claim. The first Shiigi citation refers to a java applet email client, and a java application version of an instant messaging client. Both are communications programs that run locally, unlike the claimed invention [Shiigi, there are two versions of the Handwriting Messaging Client software described below, i.e., a Java applet and a Java application version, col 4 lines 51-54]. The second Shiigi citation refers to a server component for handwriting recognition and its functioning with the client drawing and messaging software mentioned in the previous citation. There is no suggestion in Shiigi of providing a remote computer visual interface desktop to users. Likewise, as detailed in claim 132, Simonoff does not disclose a terminal-emulation thin client. In any case, neither reference teaches any element of the claim: a commercial service of access to thin client-delivered server applications having GUIs by activating a hyperlink.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. do not teach the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion in the references themselves that they be combined as the O.A. suggests.

The Rejection of Claim 165 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose said file restriction means are varied depending upon parameters of said specific session selected from the group consisting of: (a) the identity of said hyperlink recipient user; (b) the network address of said hyperlink recipient user; (c) whether a qualifying action has been performed by said recipient user; and (d) whether authentication information has been provided by said hyperlink recipient user [Shiigi, authenticate, col 8 lines 35-67].

Reification of Claim 161. Claim 165 is dependent upon claim 161, and is a reification and narrowing of that claim. It clears the requirements of USC 102 and 103 for the reasons detailed under remarks for claim 161.

Claim 201 Added to Clarify What Is Claimed. Claim 201 has been added which simply claims variable file restrictions. It clarifies that file restrictions are not limited to the list in claim 165. It is dependent on claim 161.

The Rejection of Claim 166 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose said authentication information is selected from the group consisting of: (a) a password; (b) the network address of said hyperlink recipient user; (c) a digital signature, and (d) information provided via the HTTP authentication protocol [Shiigi, authenticate, password, col 8 lines 35-67].

Claim 166 is a reification of claim 165's term "authentication information". It clears the requirements of USC 102 and 103 for the reasons detailed under remarks for claim 165.

Shiigi Authentication Refers to I.M. Account Login. The complete Shiigi citation

makes clear that he is referring to a server verifying that a user is a registered user of an Instant Messaging application, which cannot function without broadcasting the immediate availability online of participants: [Shiigi, when the client logs onto the server, the server looks up the user name and password in order to authenticate that person as a registered user, then the user is added to the list of active users, col 8 lines 42-45]. The complete citation makes clear that the authentication Shiigi is referring to is requested in order to allow a named user access to his own application account. There is no suggestion that anyone other than the name on the user account would be granted access to the application and any stored files.

Novelty. In contrast, in claim 166, the authentication information claimed is not used to allow a known user to access his server account. Rather, it is used in a computer algorithm to decide whether to grant a potentially large number of persons access to a particular file and online application, less like logging on to a user's own online account, and more like restricting access to a web site to particular persons, which is well known in the art. *What is not known* is the combination of such authentication with hyperlink access to online thin-client delivered applications and associated files.

Valuable, new, improved, and unexpected results. Employing the novel file restriction means of claim 161 in conjunction with information gleaned from or about a hyperlink activator results in unexpected results for collaboration and I.P. protection. The AppLink sender or creator has the ability to set and change conditions associated with an AppLinked file (for example, file access and usage permissions), or any other variable (for example, the degree to which the AppLink usage will be monitored), for example, to facilitate protection of the AppLink creator's intellectual property (the AppLinked file). This ability to attach various conditions or enable various features for AppLinks is a central advantage that AppLinking has over other file transmission and access systems. The settable variables may depend on other criteria as well, by which any Settable AppLink Variable may be set and varied, for example, according to the specific identity of the recipient or the nature of the file, or whether the recipient is a member of a specific

group of users, the recipient's location or country, as identified by his IP address, or any other relevant criteria.

The Rejection of Claim 183 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose notifying at least one designated user upon the occurrence of hyperlink access events selected from the group consisting of (a) hyperlink activation; (b) file access; and (c) file alteration [Shiigi, notify to client, col 8 lines 16-30].

Claim Rewritten and Divided Into Two. The claim 183 has been rewritten to claim basic notification upon hyperlink access events, and claim 202, dependent upon claim 183, has been added, reifying the hyperlink access events with the above list of a-c.

Claim 183 now reads: The method of claim 151, further comprising the step of notifying at least one designated user who is not said least one remote recipient user upon the occurrence of hyperlink access events.

Claim 202 reads: The method of claim 183, wherein said hyperlink access events are selected from the group consisting of:

- (a) hyperlink activation;
- (b) file access; and
- (c) file alteration.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. do not teach the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion in the references themselves that they be combined as the O.A. suggests.

Shiigi Citation Misinterpreted- Physical Difference With Claimed Invention. The Shiigi citation is to notify a recipient that he has received a message in real time, similar to Instant Messaging mode. Note that the notification is to the *receiver*, not the *sender*. [Shiigi, the receiving Handwriting Java Client is notified when an email message to the Real Time Jave Server, and retrieves the email message from the server. In this way, communications can take place using the Handwriting Java Client in near real time, col 8 lines 23-28] Words to clarify that the notification goes to a person other than the receiver have been added to the claim.

Shiigi functions oppositely to claimed invention. If notification about hyperlink access were sent to the person doing the accessing, it would have no useful purpose, since that person already knows he is accessing the hyperlink.

Valuable, new, improved, and unexpected results. This difference is significant and nonobvious, because the result is to inform or notify the sender or a member of his collaborative group of the accessing or alteration of an applinked file, in order, for example, to allow said user to approve or disapprove of the changes. Or, for example, a salesperson could be notified that an applink to a sales quotation has been accessed. This potentially real-time notification, and the knowledge that a prospect has read the quote, would obviously be of value if, for example, the salesperson were just about to visit the sales target. These unexpected results indicate that the novel structure of this claim is non-obvious.

The Rejection of Claim 185 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose said notification includes details about said hyperlink access events [Shiigi, notify to client, col 8 lines 16-30].

This claim is a reification and narrowing of claim 183. Please see remarks for that claim for discussion of how the citation is misinterpreted and reasons that this is claim is novel and unobvious (clears USC 102 and 103).

The Rejection of Claim 186 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose said details about said hyperlink access events comprises data selected from the group consisting of (a) details about said recipient user; (b) the time of activation of said application file hyperlink; (c) information about said file associated with said application file hyperlink; (d) the network location of said hyperlink recipient user; (e) information about any changes to said data file [Shiigi, notify to client, col 8 lines 16-30].

This claim is a reification of claim 185's term "details about said hyperlink access events". Please see remarks for that claim for discussion of how the citation is misinterpreted and reasons that this is claim is novel and unobvious (clears USC 102 and 103).

The Rejection of Claim 196 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose (g) forwarding said incoming email without the attached file to the intended member user's email account; and (h) said member user accessing said attached data file by activating said application file hyperlink [Shiigi, recipient's Email box, col 7 lines 67; web page col 7 line 54].

Incorrect Claim Referred to By O.A. The actual test of claim 196 reads:

(g) providing computer software executable code means on said computer server system adapted to select and start said file-compatible server-based computer application in a user session;

Non-Relevant Citation. Claim 196 does not mention email. Rather, it is a dependent claim of claim 151, which is an independent claim for the basic application-file hyperlink (applink) method. Claim 196 is a reification of claim 151's term "selecting said file-compatible application".

Novelty. Claim 196 is novel for reasons detailed above under claim 151.

Valuable, new, improved, and unexpected results. The use of an algorithm to select which thin-client delivered application should be used to open a file is nonobvious. A software algorithm may be implemented to select the appropriate application to open a file within an AppLink, based on appropriate criteria, which could include but is not limited to: recipient ownership of application user rights to possible application choices; any license agreements for possible application choices including but not limited to end-user agreements and service provider agreements; and application file compatibility, or which file types possible application choices can open. When an AppLink is sent to a recipient, the selection of the server-based application to open the file depends on server-based application software license terms. Many end-user software license agreements contain the limitation that it cannot be used by more than one person at a time, thus losing potential sales. If a user sends an AppLink which requires an application which has such a license, one way to comply with that license provision is to ensure that the recipient has the right to use that application also. Before the application is presented to the recipient, if the algorithm had access to an AppLink recipient application rights database, as depicted in Figure 18, it could be verified that the recipient owns or is renting the application or otherwise has rights to use the application. If the recipient did not have rights to use that application, the AppLink application selection algorithm could search for another application which had terms that allow that user to use the application. Such

terms could include applications licensed by the application service provider on a “per seat” basis, that is, not assigning rights to individual users, but, for example, to a maximum number of simultaneous users, whose identity could change. The terms might include free “demo” licensing for AppLink recipients. Another type of application which might be considered by the algorithm could be an “Open Source” application which typically has a license permitting unrestricted use.

The algorithm could also consider other applications for which the recipient has rights, and which are compatible with the file being AppLinked, but not necessarily the application used by the AppLink sender to create the file (“native application”). As a lower priority, a compatible “viewer” application with appropriate licensing could be selected, with the limitation that the user may not be able to interact with the file to the degree that he could if a fully capable native file application were used. This embodiment of the present invention, by which licensing impacts application functionality, may be implemented in conjunction with a previously-discussed embodiment, by which various application versions have different functionality in order to effect file permission security. That is, licensing terms may give differing levels of functionality for different payments to the vendor or different circumstances. A “demonstration” license might require disabled functionality.

As an example, illustrative in that many possible file access avenues are explored, assume that a user “Joe” has created a flowchart with the Microsoft® Visio® for Windows® application. He creates an AppLink to the file and sends it to recipient “Larry”. The selection algorithm would first see if Larry is on file with an Application License Tracking Database that the algorithm has access to. If Larry is found, the algorithm looks to see if he has rights to Visio, the application used to create the file. In the event that he does not, the algorithm looks to see if the Visio software vendor is perhaps allowing AppLink to be used to non-owners as a marketing mechanism. If the Visio vendor is not, the algorithm then looks for other applications that Larry owns or has rights to which are capable of opening the file. In the event that he has none, the

algorithm then looks at applications that the application service provider has licensed on a per-seat basis that are file-compatible. If the ASP has none, the algorithm then looks at open source programs that the ASP has available to it to see if any are file-compatible. If none are, the algorithm then looks to see if the ASP has any less-capable “viewer”-type applications with appropriate licenses that could be used to at least display the file to the recipient. In the event that the ASP does have such a viewer, the algorithm selects it and the AppLink Server presents the file to Larry inside the viewer. As a further implementation of this license tracking function, a database may be provided for use by an AppLink Recipient Application Selection Algorithm as described above, which contains the software applications and license terms that an AppLink recipient has rights to. An AppLink Recipient Application Selection Algorithm according to the present invention could function without knowledge of the application rights of AppLink Recipients, but access to such a database would improve the performance of such an algorithm by increasing the number and quality of the choices available to it, ensuring that the optimal legal application is used for file access.

The Rejection of Claim 197 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose (a) the provision of valuable consideration by said recipient user in exchange for rights to use available file-compatible application; (b) the provision by said recipient user of information authenticating the recipient user’s legal rights to use available file-compatible applications {Shiigi, authenticate, col 8 lines 40-45}; and (c) the subscription by said recipient user to rights to use said available file-compatible application [Simonoff, a format compatible, col 9 lines 10-29].

Shiigi Authentication Refers to I.M. Account Login. The complete Shiigi citation makes clear that he is referring to a server verifying that a user is a registered user of an Instant Messaging application, which cannot function without broadcasting the immediate availability online of participants: [Shiigi, when the client logs onto the server,

the server looks up the user name and password in order to authenticate that person as a registered user, then the user is added to the list of active users, col 8 lines 42-45]. The complete citation makes clear that the authentication Shiigi is referring to is requested in order to allow a named user access to his own application account. There is no suggestion that anyone other than the name on the user account would be granted access to the application and any stored files.

Simonoff "format compatible" Refers to Application Running on Client, Not Server. The Simonoff citation "a format compatible" does not refer to a remotely-operating *application compatible with a computer file*, as claimed. Rather, it refers to the server translating application specific message traffic into script *compatible with the client* on the user's PC. [Simonoff, "the Application Server can translate the application specific message traffic to a format compatible with the Universal Client device, i.e., GUIScript", col 9 lines 21-23].

Nowhere in Simonoff is the word "compatible" associated with a remote data file, because the Simonoff invention is application-oriented, not data file-oriented.

Reification of Claim 196. Claim 197 is a reification of claim 196's term "legal rights". It clears USC 102 and 103 for reasons detailed above for claim 196.

No suggestion to combine references. As detailed above, all the references mentioned by the O.A. do not teach the meaning that the O.A. states they have. However, even if the meaning was as the O.A. states, there is no suggestion in the references themselves that they be combined as the O.A. suggests.

Novelty. In contrast, in claim 197, the authentication information claimed is not used to allow a known user to access his server account. Rather, it is used in a computer algorithm to decide whether to grant a potentially large number of persons access to a particular file and online application, less like logging on to a user's own online account, and more like restricting access to a web site to particular persons, which is well known in the art. *What is not known* is the combination of such authentication with hyperlink

access to online thin-client delivered applications and associated files.

Valuable, new, improved, and unexpected results. Employing the novel file restriction means of claim 196 in conjunction with information gleaned from or about a hyperlink activator results in unexpected results for collaboration and I.P. protection. The AppLink sender or creator has the ability to set and change conditions associated with an AppLinked file (for example, file access and usage permissions), or any other variable (for example, the degree to which the AppLink usage will be monitored), for example, to facilitate protection of the AppLink creator's intellectual property (the AppLinked file). This ability to attach various conditions or enable various features for AppLinks is a central advantage that AppLinking has over other file transmission and access systems. The settable variables may depend on other criteria as well, by which any Settable AppLink Variable may be set and varied, for example, according to the specific identity of the recipient or the nature of the file, or whether the recipient is a member of a specific group of users, the recipient's location or country, as identified by his IP address, or any other relevant criteria.

The Rejection of Claim 198 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose (a) providing a user group comprising at least one member user with access to an email account [Shiigi, recipient's email box, col 7 lines 67; web page col 7 lines 54]; (b) intercepting all incoming emails for said user group prior to delivery to the email accounts of the at least one member user of said user group; (c) screening said incoming emails for attached data files {Simonoff, filtering or filtered, col 10 lines 41-55, 56-col 11 line 6}; (d) removing said attached data files and storing said attached data files on the said computer server system [Simonoff, embed or remove, col 15 lines 55-67]; (e) creating an application file hyperlink to said attached data files; (f) appending said application file hyperlink to an incoming email to which said attached data file had been

attached [Simonoff, program can be modified responsive to a specific scripting language, col 4 lines 34-38].

Email Accounts Well-known in Art, Claimed Combination is Not. Shiigi does refer to a user having an email account. This is well-known in the art. What is claimed, however, is a user with an email account in combination with the email gateway server, which intercepts emails before they get to a recipient and replaces the attachment with an applink.

Simonoff "filtering" Is Not Hyperlink Detection. The cited Simonoff reference "filtering or filtered" does not refer to detecting a hyperlink activation *from* a client or user. In fact, it has nothing to do with hyperlinks. Rather, it refers to selectively sending commands *to* different clients based on their different network addresses [Simonoff, the server host can selectively send GUIScript to multiple client hosts...by filtering the id_number, col 10 lines 53-55.].

Simonoff "embed or remove" Is Not a Hyperlink. The reference, 'embed or remove' does not refer to a hyperlink, or to dissasociating and reassociating a hyperlink. The full citation reads, 'the Universal Client...provides the mechanism to remove requirements for specific embedded display capabilities from any distributed system architecture'. This refers to the ability of the Universal Client to create a GUI on the client PC.

However, contrary to this citation, elsewhere Simonoff acknowledges that the GUI is not really independent of embedded display capabilities because it requires and uses a graphical library resident on the client PC [Simonoff, GUIScript can instantiate any GUI object common between Microsoft Windows, X-Windows, and the Java awl graphics library, col 11 lines 41-43]. In contrast, in the claimed invention, the client PC has no requirement to have any graphics library at all, since the GUI is generated on the server and transmitted to the client in essentially a terminal emulation mode.

Simonoff "program can be modified" Does Not Detail Claimed Combination.

Simonoff does not disclose associating a hyperlink with a specific file and server-based

application, as detailed above in the arguments for claims 151 and 132. Therefore, Simonoff does not disclose modifying the interface of said server-based application. It is the combination of said hyperlink with the modification of the server-based application's interface which is novel.

Valuable, new, improved, and unexpected results. A server screens all incoming e-mails for attachments. If an e-mail has an attached file, this file is removed from the e-mail and stored on the server after being scanned for viruses. An AppLink to the file is created and the AppLink is appended to the e-mail. The e-mail is then forwarded to its intended recipient, who opens the attachment via the AppLink, thus eliminating the danger of computer viruses for the recipient's computer. **Acute, unmet need.** Computer viruses propagated by email attachments, as is well known, cause billions of dollars of damage annually.

The Rejection of Claim 199 Is Overcome

The Office Action stated that claim 199 had similar limitations to claim 198, and was rejected for the same rationale.

Please see the remarks for claim 198, above. Claim 199 clears the requirements of USC 102 and 103 for the reasons detailed under remarks for claim 198.

Novelty. Claim 199 addresses a completely different need and functions differently from the invention of claim 198. In a related embodiment to claim 198, but for a contrasting purpose, outgoing e-mail can be treated similarly, by removing and storing the attachments, and creating an AppLink which is added to the e-mail instead.

Valuable, new, improved, and unexpected results. For outgoing e-mails, the replacement of all attachments with AppLinks may aid in the control of a company's intellectual property, by ensuring that the attached files remain with the company, and

AppLink recipient file permissions or properties may be restricted centrally according to company policy and the recipient permissions in general.

The Rejection of Claim 200 Is Overcome

The Office Action stated that Simonoff-Shiigi disclose centrally setting parameters of the file access and manipulation restrictions for said user group...as inherent features of manipulate the authentication [Shiigi, manipulate, col 2 lines 52-64].

Claim 200 does not include the list of items a thru d as quoted by the O.A.

The citation from Shiigi does not refer to authentication. However, the full citation makes clear that his meaning of the word "manipulate" is related to applications that are competely resident upon the client PC [Shiigi, ... a drawing editor/viewer that allows the user to compose, manipulate, and view handwritten or handdrawn messages, col 2 lines 58-60]. Shiigi does not mention authentication in this section because there would be no reason to authenticate a user on his own PC to use his own applications or access his own files. Further, there is no suggestion in Shiigi or Simonoff that thin client applications be used to deliver access to files.

Claim 200 is dependent on claim 199, which claims a gateway for outbound emails. The establishing and setting of security levels adds suprising results to the invention of claim 199. For outgoing e-mails, the replacement of all attachments with AppLinks may aid in the control of a company's intellectual property, by ensuring that the attached files remain with the company, and AppLink recipient file permissions or properties may be restricted centrally according to company policy and the recipient permissions in general.

The Rejection of Claim 164 Is Overcome

The Office Action stated that Simonoff discloses said file access and manipulation restrictions (i.e.: permission) are selected from the set consisting of (f) whether said data file or portions thereof may be copied onto the local computer memory clipboard of the remote recipient user computing device [Simonoff, permitting the computer to utilize a browser, col 5 lines 54-67; download of web page, col 8 lines 26-50].

Misunderstood Reference. The full Simonoff citation makes clear that Simonoff's meaning of the word "permission" is "enabling the capability", not "screening for permitted users" as in the claimed invention [Simonoff, ...a storage device...which stores computer readable instructions for generating the instructions for permitting the computer to utilize a WWW browser providing a JAVA virtual machine..to generate predetermined GUI objects, col 5 lines 56-67].

- (a) whether the data file may be accessed;
- (b) a number of times the data file may be accessed [Felciano, tracking access, col 2 lines 20-48];

Non-Relevant Reference. This Felciano citation addresses tracking a user's activity, not the claimed limiting of that activity.

- (c) a particular time period during which the file may be accessed [Felciano, perform periodic web searches, col 7 lines 43-67];

This is a misunderstood reference. Felciano teaches automatically performing web searches periodically after a user session based on information gleaned from that user session, in hopes of finding something the user will be interested in. It does not refer to restricting the time of a user session.

- (d) whether said hyperlink recipient user may print (i.e.: view) the data file locally [Felciano, viewing the page, col 5 lines 35-42];

Felciano teaches downloading a tracking applet which monitors activities while a user is viewing a page (things like mouse cursor movement, etc.). This applet has no effect on

the user's access to a page or to what he can do with the web page data. It is simply a monitoring (spyware) program. In contrast, the claimed invention is for limiting access to a server-based file, not monitoring that access.

(e) whether said hyperlink recipient user may save said first data file locally onto said remote recipient user computing device [Felciano, storing and tracking, col 4 lines 37-65];

The Felciano citation teaches a method for tracking various data about a user's web sessions. In contrast, the claimed invention is for limiting access to a server-based file, not monitoring that access.

(f) whether said data file or portions thereof may be copied onto the local computer memory clipboard of the remote recipient user computing device;

(g) whether, after alteration by said recipient user, said data file or portions thereof may be saved onto said computer server system, whereby the original copy of said data file is replaced by the altered file, and said application file hyperlink is now associated with said altered file, and (h) whether, after alteration by said recipient user, said data file or portions thereof may be saved onto said computer server system, whereby the original copy of said data file is not replaced by the altered file and remains on the storage system of said server system [Felciano, permits task to be performed, original file or URL, modified URL, col 4 lines 21-35].

Fundamentally, there is no suggestion in Felciano of a recipient user altering anything, unlike what is claimed. The citation "permits tasks to be performed" refers to tasks accomplished by the program/server, not the user.

The O.A. states that it was well known that Internet information can be monitored (i.e.: customer, visitor hits) or print or stored on local memory (i.e.: saving downloaded file) or edit file by recipient as taught by Felciano [Felciano, perform periodic web searches, col 7 lines 43-67; viewing the page, col 5 lines 35-42; storing and tracking, col 4 lines 37-65; permits task to be performed, original file or URL, modified URL, col 4 lines 21-35]. Therefore it would have been obvious to an ordinary skill in the art to incorporate the technique of modifying web session as

taught by Felciano into Simonoff's apparatus in order to utilize the applet. Doing so would provide the Web client a dynamic tool to modify hypertext browsing activities.

Felciano does not refer anywhere to a user editing a file. The citation "perform periodic web searches" refers to automatically performing web searches periodically after a user session based on information gleaned from that user session, in hopes of finding something the user will be interested in. It does not refer to restricting the time of a user session. "storing and tracking" teaches a method for tracking various data about a user's web sessions. In contrast, the claimed invention is for limiting access to a server-based file, not monitoring that access. "permits task to be performed" refers to tasks accomplished by the program/server, not the user. Fundamentally, there is no suggestion in Felciano of a recipient user altering anything, unlike what is claimed.

Felciano explicitly discloses that his invention cannot monitor application usage, unlike the claimed invention [Felciano, Lamprey cannot tack subsequent interactions with the FTP service... This limitation is due to the fact that the client browser is now interacting through the FTP protocol and not the HTTP protocol, col 6 lines 47-49]. For the same reasons, Felciano cannot track or monitor application usage in the claimed invention, which uses the thin client protocol, not HTTP.

As mentioned above under the remarks for claim 132, Simonoff does not teach a thin client as claimed.

The Rejections of Claims 167-169 Are Overcome

The Office Action stated that Simonoff-Felciano discloses said qualifying action is accepting an agreement; said agreement obligates the recipient user to limit disclosure of the content of said data file; said agreement is a license agreement {Felciano, a simple rule, col 4 lines 1-8}.

Incorrect Reference. The citation does not contain the words "a simple rule".

Reification of claim 166. Claims 167-169 are dependent upon claim 166 and reifications of that claim. They meet the requirements of USC 102 and 103 for reasons detailed under the remarks for that claim.

Other References Not Relied Upon By the O.A.

The O.A. stated that the prior art made of record and not relied upon is considered pertinent to the applicant's disclosure. USP 6,256,620 B1 Jawahar et al discloses method and apparatus for monitoring application information on web pages access. Applicant has examined the reference and does not find any relevant limitations.

Concluding Remarks

Commercial Success. Applicant has achieved commercial success that flows from the functions and advantages disclosed in the specification. A declaration thereto with supporting documents was attached to the previous amendment B. The Examiner is invited to peruse the Applicant's company's web site, www.marix.com, which details commercial implementations of the claimed inventions.

From MPEP 716.03(b): In *ex parte* proceedings before the Patent and Trademark Office, an applicant must show that the claimed features were responsible for the commercial success of an article if the evidence of nonobviousness is to be accorded substantial weight.

Applicant has implemented the claimed invention in hardware and software. Applicant has achieved commercial success by selling the invention to numerous customers as laid out in the declaration attached to the previous Amendment.

Nexus Between the Claimed Invention and Evidence of Commercial Success. The commercial success achieved is directly attributable to the claimed invention. This is clearly shown by the fact that the commercial product as sold uses a separate product to carry out the web-enablement of the server-based applications, i.e., the Tarantella web-enabling program (an equivalent to Citrix). That is, it is an add-on package which adds AppLink capability to the functionality of the basic web-enabling package. There would be no reason to purchase it if the customer did not want the additional functionality.

Example: One customer, an auto dealership, uses AppLink to deliver price quotes to potential customers while retaining the quotes on its server to prevent the customer from simply taking the quote to another dealer to get a better price.

Example: The US Air Force Academy bought the claimed invention to distribute extremely sensitive information to its teachers. Specifically, there is a list of potentially suicidal students that is distributed to the teachers so that they can "go easy" on those students. This list was inadvertently emailed to local news organizations. The sender realized her error immediately, but the information was irretrievably on its way. With AppLink, the URL can be inactivated at any time, and further is always linked to the most current document.

Commercial Acquiescence. Applicant has demonstrated commercial acquiescence by licensing the present invention's technology to Tarantella, a division of Sun Microsystems, as laid out in the declaration with supporting documents attached to the previous Amendment B.

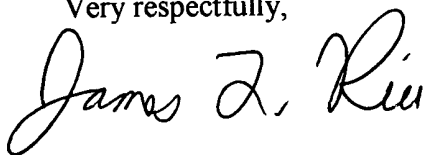
Invention Has Been Examined Previously by the USPTO. This invention was submitted as an international patent application (International PCT Application No. PCT/00/24719, System and Method of Permissive Data Flow, filed September 8, 2000), which contained claims broader than the ones in the present application. This application was examined by Examiner James R. Matthews as the US agent of the PCT. In an International Preliminary Examination dated October 3, 2002. Almost every claim was found to be patentable by Examiner Matthews.

Invitation To Try a Commercial Model of the Invention. The Examiner is invited to try out the claimed invention from the comfort of his workplace or home. Actually trying the invention briefly can quickly illustrate its novel features and advantages. The Applicant has created a web site with numerous example AppLinks which are hosted on the Applicant's company servers. The web site is: <http://applinks.blogspot.com>. *Please note the client PC requirements as detailed on the web page.* Please feel free to call the Applicant at any time for assistance, if needed, at 612 963-8233. The Applicant will know within seconds when the Examiner has accessed the hyperlinks because an email with access details is automatically sent to the hyperlink originator, as claimed for the present invention. Also, the Examiner is invited to try an AppLink creation account, wherein the Examiner can upload his own documents to the AppLink server and create his own AppLinks. Also, the Examiner is invited to try out an account on the Applicant's company's demo AppLink Email Gateway Server which screens incoming emails and substitutes an AppLink for any attached files.

For all of the above reasons, applicant submits that the specification and claims are now in proper form and that the claims all define patentability over the prior art. Therefore applicant submits that this application is now in condition for allowance, which action he respectfully solicits.

If the application is not believed to be in full condition for allowance, applicant respectfully requests assistance and suggestions of the Examiner.

Very respectfully,

A handwritten signature in cursive script that reads "James L. Rice".

James L. Rice

Applicant Pro Se

2115 Penn Ave S.
Minneapolis, MN 55405
Tel 612 963-8233

Certificate of mailing: I certify that this correspondence, and attachments, if any, will be deposited with the United States Postal Service by First Class Mail, postage prepaid, in an envelope addressed to "Box Non-Fee Amendments, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" on the date below.

Date: 6 SEPT, 2005

Inventor's Signature: James Z. Riu

CLEAN COPY OF CLAIMS: *The following is a list of all claims in the application with their status and the text of all active claims, in clean copy format.*

1.-131. (CANCELED)

132. (CURRENTLY AMENDED) A method to allow a user of a local computer to access a remote computer file with a remote application, the method comprising the steps of:

- (a) providing a computer server system connected to a communications network;
- (b) the server system detecting the activation by the user of a hyperlink associated with the computer file;
- (c) operating on said computer server system an application program which is compatible with or capable of loading and operating upon said computer file, and has a graphical user interface;
- (d) opening the computer file in the application program; and
- (e) operating a thin client of the terminal emulation type on the local computer, the thin client allowing the user to provide input to the application program running on the remote computer, and

whereby control and protection of the computer file is retained, while providing simple and easy access to the user via said server-based applications delivered via thin client.

Claim 133. (CURRENTLY AMENDED) The method of claim 132, wherein the hyperlink points to a network location or address associated with metadata identifying the computer file.

Claim 134. (PREVIOUSLY PRESENTED) The method of claim 132, wherein the application program is associated with the computer file after the activation of the hyperlink is detected.

Claim 135. (PREVIOUSLY PRESENTED) The method of claim 132, wherein multiple application programs are available to be associated with the computer file, and the associated application program is selected based on selection criteria chosen from the group comprising:

- (a) legal rights of the user to use the application programs;
- (b) capabilities of the application programs; and
- (c) properties that are associated with the hyperlink.

Claim 136. (CURRENTLY AMENDED) A method for handling incoming e-mails at an e-mail gateway comprising the steps of:

- (a) identifying a data file attachment on an incoming e-mail addressed to a recipient;
- (b) creating a copy of the identified attachment on a storage device accessible by the e-mail gateway;
- (c) creating a reference or hyperlink associated with the attachment copy;
- (d) deleting the attachment from the incoming e-mail and adding said hyperlink to the email;
- (e) operating on a remote computer remote from the recipient an application program associated with and compatible with the attachment copy;
- (f) opening the attachment copy in the application program running on the computer remote from the recipient upon activation of the hyperlink; and
- (g) operating a thin client on a local computer used by the recipient, the thin client allowing the recipient to provide input to and receive output from the application program running on the remote computer,

whereby the email attachment cannot pose a virus danger to said local computer since it is manipulated remotely, and whereas the local computer has no need of a compatible program to open said attachment.

Claim 137. (CURRENTLY AMENDED) A method for handling outgoing e-mails at an e-mail gateway comprising the steps of:

- (a) identifying a data file attachment on an outgoing e-mail addressed to a recipient;
- (b) creating a copy of the identified attachment on a storage device accessible by the e-mail gateway;
- (c) creating a hyperlink associated with the attachment copy;
- (d) replacing the identified attachment in the outgoing e-mail with the hyperlink;
- (e) forwarding the outgoing e-mail with the hyperlink to the recipient;
- (f) detecting an activation of the hyperlink by the recipient;
- (g) operating on a remote computer remote from the recipient an application program compatible with said attachment copy or file;
- (h) opening said attachment copy in said application program; and
- (i) operating a thin client on a local computer used by the email recipient, the thin client allowing the recipient to provide input to and receive output from said application program,

whereby control and protection of outgoing data attachments is retained by removing said data from outgoing emails, while giving appropriate access to recipients via server-based applications delivered via thin client.

Claim 138. (PREVIOUSLY PRESENTED) The method of claim 137, further comprising the step of imposing file access restrictions on said attachment copy.

Claim 139. (PREVIOUSLY PRESENTED) The method of claim 207, further comprising the step of setting a variable level of file access restrictions based upon a security level associated with the recipient, whereby attachment access can be varied according to the trust placed in a particular recipient individual as represented by said security level.

Claim 140. (PREVIOUSLY PRESENTED) The method of claim 207, further comprising the step of setting a variable level of access restrictions based upon a security level associated with the identified attachment, whereby attachment access can be varied according to the sensitivity of a particular attached document.

Claim 141. (CURRENTLY AMENDED) An application server comprising:

- (a) a network interface to communicate over a computer network;
- (b) at least one processing unit for running software programs and components;
- (c) an application remote-operation enabling program capable of operating an application program which has a GUI and transmitting the GUI of said application program in a terminal emulation mode to a thin client operating on a remote computer, and providing remote interaction with the application program via the thin client over the computer network;
- (d) a storage device containing a computer file;
- (e) an association component that associates a hyperlink with the computer file;
- (f) a listener component that detects an activation of the hyperlink over the computer network by the remote computer; and

- (g) an initiator component that, upon detection of the hyperlink activation, opens the computer file in said application program and transmits the graphical user interface of said application program to the thin client operating on the remote computer,

whereby control and protection of the computer file is retained, while giving appropriate access to recipients via said server-based applications delivered via thin client.

Claim 142. (PREVIOUSLY PRESENTED) The application server of claim 141, wherein the at least one processing unit is selected from the group consisting of:

- (a) a single server;
- (b) a personal computer; and
- (c) multiple, separate computers operating as a single, logical server.

Claim 143. (CURRENTLY AMENDED) A method comprising the steps of:

- (a) providing a computer server system which is connected to a communications network and which contains executable code of at least one server-based computer application which operates natively with a graphical user interface,
- (b) providing an application hyperlink that is associated with said server-based application,
- (c) providing a computing device to at least one recipient user which is connected to said communications network,
- (d) providing computer software executable code on the recipient user's computing device which ~~for~~ activates said application hyperlink,
- (e) providing computer software executable code on said computer server system which detects the activation of the hyperlink by said recipient user's computing device, and

- (f) providing computer software executable code on said computer server system which selects and launches said server-based computer application,
- (g) providing computer software executable code on said computer server system which transmits the graphical user interface of said server-based computer application to said recipient user's computing device in a terminal emulation mode, and receives inputs to said graphical user interface from the user; and
- (h) providing thin client computer software executable code on said recipient's computer which receives and displays said graphical user interface from said computer server system, and transmits the inputs of said recipient user back to the server-based application,

whereby a wide variety of potentially anonymous users are provided simple and easy access to said server-based applications delivered via thin client, irrespective of the operating system and the overall capabilities of the recipient's computer, and allowing effortless access to virtually any application to be placed anywhere a hyperlink can be placed, such as an email or web page.

Claim 144. (CURRENTLY AMENDED) The method of claim 143, comprising the additional steps of:

- (a) providing computer software executable code on said computer server system for creating said application hyperlink; and
- (b) transmitting said application hyperlink to said at least one recipient user,

whereby a wide variety of potentially anonymous users are provided simple and easy access to said server-based communications applications delivered via thin client, irrespective of the operating system and the overall capabilities of the recipient's computer, and allowing effortless access to virtually any communications application to be placed anywhere a hyperlink can be placed, such as an email or web page.

Claim 145. (CURRENTLY AMENDED) The method of claim 143, wherein said server-based computer application comprises a first communications program or application.

Claim 146. (CURRENTLY AMENDED) The method of claim 145, comprising the additional steps of:

- (a) providing a second communications program or application to one or more additional users which are compatible with or capable of communicating with said first communications program; and
- (b) providing address configuration computer executable code on said server system which configure said first communications means with the communications address of said one or more additional users,

whereby a wide variety of potentially anonymous users are provided simple and easy access to said server-based communications applications delivered via thin client, irrespective of the operating system and the overall capabilities of the recipient's computer, and allowing effortless access to virtually any communications application to be placed anywhere a hyperlink can be placed, such as an email or web page, and allowing easy communication with said second communications means because of the adding of an address, potentially of the sender, to said first communications means.

Claim 147. (CURRENTLY AMENDED) The method of claim 146, wherein said first and second communications program or application comprises programs selected from the group consisting of:

- (a) an email program;
- (b) an instant messaging program;

- (c) a voice-over-internet-protocol program;
- (d) a video conferencing program; and
- (e) an internet relay chat application.

Claim 148. (PREVIOUSLY PRESENTED) The method of claim 212, wherein said server-based application comprises a computer visual interface desktop work area, and further comprising the steps of:

- (a) providing one or more native software server-based applications and
- (b) providing access to or links to said one or more native software server-based applications within said computer visual interface desktop work area,

whereby a wide variety of potentially anonymous users are provided simple and easy access to said server-based applications delivered via thin client, irrespective of the operating system and the overall capabilities of the recipient's computer, and allowing effortless access to virtually any application to be placed anywhere a hyperlink can be placed, such as an email or web page.

Claim 149. (PREVIOUSLY PRESENTED) The method of claim 217, wherein said native software applications comprise applications selected from a group consisting of:

- (a) an email program,
- (b) an instant message program,
- (c) stock quote program,
- (d) news delivery program,
- (e) weather delivery program,

- (f) a group scheduling program,
- (g) a data file storage program; and
- (h) a calendar program.

Claim 150 (PREVIOUSLY PRESENTED) The method of claim 148 used in conjunction with a commercial service, further comprising the step of:

providing access to said computer visual interface desktop to subscribers of said commercial service, whereby said application hyperlink recipient may utilize said visual interface desktop in a manner similar to accessing a personalized home web page which utilizes html or javascript code, but with the full functionality of standard software applications instead of the limited functionality of said html or javascript code.

Claim 151. (CURRENTLY AMENDED) A method to allow remote users to access computer files with server-based applications by activating a hyperlink, the method comprising the steps of:

- (a) providing a computer server system connected to a communications network, the server system having access to at least one computer file and to executable code for at least one file-compatible server-based application which operates natively with a graphical user interface, the at least one application being adapted to open said at least one computer file;
- (b) providing an application file hyperlink that is associated with said at least one computer file;
- (c) providing at least one remote recipient user computing device to at least one hyperlink recipient user which is connected to said communications network;

- (d) providing computer software executable code on the at least one remote recipient user computing device for activating said application file hyperlink over said communications network;
- (e) providing computer software executable code on said computer server system which detects the activation of the hyperlink by said hyperlink recipient user over said communications network;
- (f) providing a thin client application or program on said at least one remote recipient user computer adapted to receive and display said graphical user interface of said file-compatible server-based computer application in a terminal emulation mode, and accept and transmit inputs of said user back to the server-based application;
- (g) providing computer software executable code on said computer server system adapted to select and start said file-compatible server-based computer application in a user session;
- (h) providing computer software executable code on said computer server system which opens said computer file in said file-compatible server-based computer application; and
- (i) providing computer software executable code on said computer server system for transmitting the graphical user interface of said file-compatible server-based computer application to the at least one remote recipient user computer in a terminal emulation mode and to receive inputs to said interface from the at least one hyperlink recipient user.

Claim 152. (CURRENTLY AMENDED) The method of claim 151, further comprising the step of providing computer software executable code for creating said application file hyperlink and associating the hyperlink with said at least one computer file.

Claim 153. (CURRENTLY AMENDED) The method of claim 151, further comprising the step of transmitting said application file hyperlink to said at least one hyperlink recipient user;

Claim 154. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said communications network comprises an internet.

Claim 155. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said communications network comprises a local area network.

Claim 156. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said hyperlink comprises a uniform resource locator.

Claim 157. (PREVIOUSLY PRESENTED) The method of claim 151, further comprising the steps of:

- (a) disassociating said application file hyperlink from said at least one computer file; and
- (b) associating said application file hyperlink with at least one different computer file.

Claim 158. (PREVIOUSLY PRESENTED) The method of claim 151, additionally comprising the step of modifying the application interface capability function settings of said file-compatible server-based computer application.

Claim 159. (PREVIOUSLY PRESENTED) The method of claim 151, wherein the recipient user computer is of a type selected from the group consisting of:

- (a) a personal computer;
- (b) a personal digital assistant;
- (c) a web or internet appliance device; and
- (d) a television set-top box.

Claim 160. (CURRENTLY AMENDED) The method of claim 152, wherein the graphical user interface of said hyperlink generation computer software executable code comprises a web page form.

Claim 161. (CURRENTLY AMENDED) The method of claim 151, further comprising the step of providing computer software executable code on said computer server system which restricts said recipient user's access to and manipulation of said at least one computer file.

Claim 162. (CURRENTLY AMENDED) The method of claim 161, wherein said at least one file-compatible server-based application is partially disabled.

Claim 163. (CURRENTLY AMENDED) The method of claim 161, wherein the comprises a partially disabled thin-client application.

Claim 164. (PREVIOUSLY PRESENTED) The method of claim 161, wherein said file access and manipulation restrictions are selected from the set consisting of:

- (a) whether the data file may be accessed;
- (b) a number of times the data file may be accessed;
- (c) a particular time period during which the file may be accessed;
- (d) whether said hyperlink recipient user may print the data file locally;
- (e) whether said hyperlink recipient user may save said first data file locally onto said remote recipient user computing device;
- (f) whether said data file or portions thereof may be copied onto the local computer memory clipboard of the remote recipient user computing device;
- (g) whether, after alteration by said recipient user, said data file or portions thereof may be saved onto said computer server system, whereby the original copy of said data file is replaced by the altered file, and said application file hyperlink is now associated with said altered file, and

- (h) whether, after alteration by said recipient user, said data file or portions thereof may be saved onto said computer server system, whereby the original copy of said data file is not replaced by the altered file and remains on the storage system of said server system.

Claim 165. (PREVIOUSLY PRESENTED) The method of claim 161, wherein said file restriction means are varied depending upon parameters of said specific session selected from the group consisting of:

- (a) the identity of said hyperlink recipient user;
- (b) the network address of said hyperlink recipient user;
- (c) whether a qualifying action has been performed by said recipient user; and
- (d) whether authentication information has been provided by said hyperlink recipient user.

Claim 166. (PREVIOUSLY PRESENTED) The method of claim 165, wherein said authentication information is selected from the group consisting of:

- (a) a password;
- (b) the network address of said hyperlink recipient user;
- (c) a digital signature, and
- (d) information provided via the HTTP authentication protocol.

Claim 167. (PREVIOUSLY PRESENTED) The method of claim 165, wherein said qualifying action is accepting an agreement.

Claim 168. (PREVIOUSLY PRESENTED) The method of claim 167, wherein said agreement obligates the recipient user to limit disclosure of the content of said data file.

Claim 169. (PREVIOUSLY PRESENTED) The method of claim 167, wherein said agreement is a license agreement.

Claim 170. (PREVIOUSLY PRESENTED) The method of claim 153, wherein said hyperlink is transmitted using means selected from the following group:

- (a) an email message;
- (a) an instant messaging application;
- (b) a hyperlink embedded in a web page;
- (c) a telephone;
- (d) an appropriately formatted document or file; and
- (e) a pager.

Claim 171. (PREVIOUSLY PRESENTED) The method of claim 151, comprising the additional steps of:

- (a) providing executable code on said computer server system for producing a computer visual interface desktop work area,
- (b) providing access to or links to said file-compatible application on the visual interface desktop,
- (c) providing access to or links to zero or more additional applications on the visual interface desktop,
- (d) providing access to or links to said at least one computer file or document on the visual interface desktop; and

wherein said thin client means on said at least one remote recipient computer are adapted to receive and display said computer visual interface desktop work area.

Claim 172. (PREVIOUSLY PRESENTED) The method of claim 151, additionally comprising the steps of:

- (a) after detecting said hyperlink activation, assigning a guest account to said recipient user;
and
- (b) creating a copy of said at least one computer file in the guest account; and
- (c) opening the file copy in said file-compatible server-based application instead of the original computer file.

Claim 173. (CURRENTLY AMENDED) The method of claim 172, wherein the guest account is created after detecting said hyperlink activation.

Claim 174. (CURRENTLY AMENDED) The method of claim 172, further comprising the steps of:

- (a) deleting said guest account upon the closing of the computer file by the user, or
- (b) deleting or cleaning data items from said guest account and returning said guest account to available status upon the closing of the computer file by the user.

Claim 175. (PREVIOUSLY PRESENTED) The method of claim 172, further comprising the step of predefining a plurality of guest accounts, and further wherein one of the predefined guest accounts is assigned to the user after hyperlink activation.

Claim 176. (CURRENTLY AMENDED) The method of claim 172, wherein said guest account is unassigned or returned to available status upon the closing of the computer file by the user.

Claim 177. (CURRENTLY AMENDED) The method of claim 151, wherein said at least one computer file comprises a first computer file and at least one additional computer file, and wherein said application file hyperlink is associated with both the first and the additional computer files, and comprising the additional steps of:

- (a) operating on a remote computer a plurality of application programs, said plurality of programs having at least one program compatible with or capable of opening each of said computer files;

- (b) opening the computer files in said plurality of application programs; and
- (c) transmitting the graphical user interface of said plurality of application programs to said thin client means in a terminal emulation mode.

Claim 180. (CURRENTLY AMENDED) The method of claim 151, further comprising the step of providing means for at least one additional user using at least one additional remote user computing device to view and optionally interact with the interface of said file-compatible server-based computer application in a terminal emulation mode, whereby the additional user and the first hyperlink recipient user may engage in simultaneous collaboration over said file or document.

Claim 181. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said method is provided by an entity as a service and further comprising the step of providing subscription means associated with said service whereby a remote hyperlink recipient user who is not a subscriber of said entity is solicited to subscribe to said service.

Claim 182. (PREVIOUSLY PRESENTED) The method of claim 151, wherein a log of accesses to said computer file is kept on said computer server system.

Claim 183. (CURRENTLY AMENDED) The method of claim 151, further comprising the step of notifying at least one designated user who is not said least one remote recipient user upon the occurrence of hyperlink access events.

Claim 184. (PREVIOUSLY PRESENTED) The method of claim 183 93, wherein said notification utilizes transmission means selected from the group consisting of:

- (a) an email application;
- (b) an instant message application;
- (c) a telephone; and
- (d) a pager.

Claim 185. (PREVIOUSLY PRESENTED) The method of claim 183, wherein said notification includes details about said hyperlink access events.

Claim 186. (PREVIOUSLY PRESENTED) The method of claim 185, wherein said details about said hyperlink access events comprises data selected from the group consisting of:

- (a) details about said hyperlink recipient user;
- (b) the time of activation of said application file hyperlink;
- (c) information about said file associated with said application file hyperlink;
- (d) the network location of said hyperlink recipient user;
- (e) information about any changes to said data file.

Claim 187. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said computer server system comprises a computing machine configuration selected from the group consisting of:

- (a) a single server;
- (b) a personal computer.
- (c) multiple, separate computers operating as a single, logical server.

Claim 188. (PREVIOUSLY PRESENTED) The method of claim 151, wherein the computer software executable code which detects the execution of the hyperlink by said hyperlink recipient user operates on a computing device physically separate from the remainder of said server system but connected through said communications network.

Claim 189. (old claim 189) The method of claim 187, wherein one of said multiple, separate computers operating as a single, logical server comprises a computer operating as an application server.

Claim 190 (CANCELLED)

Claim 191. (PREVIOUSLY PRESENTED) The method of claim 151, further comprising the step of downloading the thin client from the server system to said at least one remote recipient user computing device .

Claim 192. (PREVIOUSLY PRESENTED) The method of claim 191, wherein the step of downloading the thin client occurs only after said at least one remote recipient user computing device is examined and found not to already have the thin client available.

Claim 193. (CURRENTLY AMENDED) The method of claim 151, wherein the step of opening the computer file occurs only after a determination is made that the hyperlink associated with the computer file is still active or enabled.

Claim 194. (PREVIOUSLY PRESENTED) The method of claim 151, wherein changes are made to the computer file after the hyperlink is created, and the hyperlink remains associated with the latest version of the computer file.

Claim 195. (PREVIOUSLY PRESENTED) The method of claim 151, wherein the application program is not operating on the remote computer until after the activation of the hyperlink is detected.

Claim 196. (PREVIOUSLY PRESENTED) The method of claim 151, wherein said means for selecting said file-compatible application is adapted to make said selection after considering criteria selected from the group consisting of:

- (a) information about optimum applications for viewing of said file;
- (b) information about optimum applications for manipulation of said file;
- (c) said recipient user's legal rights to use available file-compatible applications; and
- (d) usage permissions associated with said available file-compatible applications.

Claim 197. (PREVIOUSLY PRESENTED) The method of claim 196, wherein the recipient user's legal rights are established by criteria selected from the group consisting of:

- (a) the provision of valuable consideration by said recipient user in exchange for rights to use available file-compatible application;
- (b) the provision by said recipient user of information authenticating the recipient user's legal rights to use available file-compatible applications; and
- (c) the subscription by said recipient user to rights to use said available file-compatible application.

Claim 198. (PREVIOUSLY PRESENTED) The method of claim 161, further comprising the steps of:

- (a) providing a user group comprising at least one member user with access to an email account;
- (b) intercepting all incoming emails for said user group prior to delivery to the email accounts of the at least one member user of said user group;
- (c) screening said incoming emails for attached data files,
- (d) removing said attached data files and storing said attached data files on the said computer server system;
- (e) creating an application file hyperlink to said attached data files;
- (f) appending said application file hyperlink to an incoming email to which said attached data file had been attached;
- (g) forwarding said incoming email without the attached file to the intended member user's email account; and

- (h) said member user accessing said attached data file by activating said application file hyperlink.

Claim 199. (PREVIOUSLY PRESENTED) The method of claim 161, further including the steps of:

- (a) providing a user group consisting of at least one member user with access to an email account;
- (b) automatically intercepting all outgoing emails for said user group;
- (c) automatically screening said outgoing emails for attached data files;
- (d) automatically removing the attached data file from the outgoing email and storing said attached file on the server;
- (e) said computer server system creating an application file hyperlink to said attached files;
- (f) said computer server system appending said hyperlink to said outgoing email;
- (g) said server forwarding said incoming email without the attached file to its intended recipient's email account; and
- (h) said intended recipient accessing said attached file by activating said application file hyperlink.

Claim 200. (PREVIOUSLY PRESENTED) The method of claim 198, further comprising the step of centrally setting parameters of the file access and manipulation restrictions for said user group.

Claim 201. (NEW) The method of claim 161, wherein the recipient user's file access and manipulation restrictions are variable.

Claim 202. (NEW) The method of claim 183, wherein said hyperlink access events are selected from the group consisting of:

- (a) hyperlink activation;
- (b) file access; and
- (c) file alteration.

Claim 203 (NEW) The method of claim 187, wherein one of said multiple, separate computers operating as a single, logical server comprise a computer operating as a file server.